

**LAPORAN AKHIR
PENELITIAN DIPA KOPERTIS WILAYAH V**



**JUDUL PENELITIAN :
IMPLEMENTASI SIMULATOR *MININET* SEBAGAI PENDUKUNG
JARINGAN KOMPUTER MASA DEPAN BERBASIS *SOFTWARE*
DEFINED NETWORK DI STMIK AKAKOM**

Dibiayai oleh :
Dana Penelitian DIPA Kopertis V Tahun anggaran 2014
Kopertis Wilayah V
Daerah Istimewa Yogyakarta
Sesuai dengan Surat Perjanjian Pelaksanaan Bantuan Penelitian Tahun Anggaran 2014
Nomor : 2174/K6/KM/2014

Tahun ke 1 dari rencana 1 tahun
Ketua/ Anggota Tim
Yagus Cahyadi S.T., M.Eng
NIDN - 0518128101
Agung Budi Prasetyo S.Kom., M.Kom
NIDN - 0003087106

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
AKAKOM
YOGYAKARTA
Nopember 2014**

HALAMAN PENGESAHAN

Judul : Implementasi Simulator *Mininet* Sebagai Pendukung Jaringan Komputer Masa Depan berbasis *Software Defined Network* di STMIK AKAKOM.

Peneliti/ Pelaksana

Nama Lengkap : Yagus Cahyadi S.T., M.Eng.
NIDN : 0518128101
Jabatan Fungsional : Tenaga Pengajar
Program Studi : Sistem Informasi
Nomor HP : 085725720897
Alamat Surel (e-mail) : yagus.cahyadi@akakom.ac.id

Anggota (1)

Nama Lengkap : AGUNG BUDI PRASETYO, S.Kom., M.Kom.
NIDN : 0003087106
Perguruan Tinggi : STMIK AKAKOM

Pembimbing

Nama Lengkap : L.N.Harnaningrum S.Si., M.T.
NIDN : 0513057101
Perguruan Tinggi : STMIK AKAKOM

Tahun Pelaksanaan : Tahun ke 1 dari rencana 1 tahun

Biaya Tahun Bejalan : Rp.4.000.000,00

Biaya Keseluruhan : Rp.4.000.000,00

Yogyakarta, 10 Nopember 2014

Mengetahui
Dosen Pembimbing



L.N.Harnaningrum S.Si., M.T
961084/0513057101

Ketua Peneliti,



Yagus Cahyadi S.T., M.Eng
121175/0518128101

Mengetahui
Ketua Program Studi



Erna Hudianti P S.Si., M.Si
961082/0528097101

Menyetujui,
Ketua Lembaga Penelitian



Dra. Syamsu Windarti, M.T., Apt
NIP/NIK 1966071019932001

DAFTAR ISI

Halaman Judul	i
Halaman Pengesahan	ii
Daftar isi	iii
Abstrak	iv
 BAB I PENDAHULUAN	 1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	2
1.5 Manfaat Penelitian	2
1.6 Target Luaran	2
 BAB II TINJAUAN PUSTAKA	 3
2.1 Teknologi jaringan komputer	3
2.2 <i>Software Defined Network (SDN)</i>	3
2.3 <i>OpenFlow</i>	4
2.4 <i>Network simulator</i>	5
2.4.1 <i>NS3</i>	5
2.4.2 <i>Packet Tracer</i>	6
2.4.3 <i>Mininet</i>	6
 BAB III METODE PENELITIAN	 8
3.1 Metode penelitian	8
3.2 Infrastruktur Jaringan STMIK AKAKOM	9
3.3 Analisis Dan Desain	12
3.4 Implementasi	14
3.5 Simulasi	17
 BAB IV HASIL PENELITIAN	 18
4.1 Hasil penelitian	18
4.1.1 Semua vlan terhubung.....	18
4.1.2 Pengujian Bandwidth ISP dan Akamai Google	21
4.1.3 Pengujian Bandwidth di client	22
4.1.4 Jaringan Laboratorium	23
4.2 Pengujian bandwidth client	24
 BAB V KESIMPULAN DAN SARAN	 26
5.1 Kesimpulan	26
5.2 Saran	26
 DAFTAR PUSTAKA	 27
LAMPIRAN - LAMPIRAN	28

Abstrak

Saat ini teknologi jaringan komputer mengalami fase kritis. Hal ini disebabkan boomingnya teknologi mobile, cloud computing, big data, internet of things, video streaming dan voice. Semua teknologi tersebut membutuhkan bandwidth yang besar dan adaptif. Jika infrastruktur jaringan saat ini (model tree/client server) masih dipertahankan, maka proses adopsi dari semua teknologi tersebut akan terkendala. *OpenFlow* adalah protokol yang digunakan untuk memisahkan antara *data plane* dengan *control plane*. Dengan protokol ini jaringan komputer menjadi tidak tergantung pada *vendor*, bersifat terbuka dan jaringan menjadi lebih dinamis yang memungkinkan untuk memproses semua teknologi baru tersebut di atas.. *Mininet* adalah simulator yang dapat mengimplementasikan protokol *OpenFlow* dan hasil dari simulasi bisa langsung diimplementasikan ke infrastruktur jaringan real. Tujuan dari penelitian ini adalah membangun lingkungan *testbed* jaringan komputer masa depan yang berbasis *Software Defined Network(SDN)* dengan protokol *OpenFlow* sehingga mahasiswa dan dosen yang ada di lingkungan STMIK AKAKOM khususnya dan peneliti luar umumnya dapat melakukan penelitian di bidang jaringan komputer masa depan berbasis *Software Defined Network* beserta dengan implementasinya.

Kata kunci: Jaringan komputer, *OpenFlow*, *SDN*, *Mininet*

BAB I

PENDAHULUAN

1.1 Latar Belakang

Saat ini teknologi jaringan komputer mengalami fase kritis. Hal ini disebabkan boomingnya teknologi mobile, cloud computing, big data, internet of things, video streaming dan voice. Semua teknologi tersebut membutuhkan bandwidth yang besar dan adaptif. Jika infrastruktur jaringan saat ini (model tree/client server) masih dipertahankan, maka proses adopsi dari semua teknologi tersebut akan terkendala.

Untuk itu diperlukan teknologi yang dapat menjadikan teknologi jaringan komputer menjadi lebih dinamis. *Open Network Foundation (ONF)* [7] mulai tahun 2011 mengembangkan protokol *OpenFlow*. Protokol ini berfungsi untuk membuat jaringan komputer menjadi terbuka sehingga jaringan komputer lebih dinamis dan tidak tergantung pada *vendor*.

Untuk bisa mengimplementasikan protokol *OpenFlow* diperlukan biaya investasi yang besar di awal. Sehingga dimungkinkan menggunakan simulator untuk proses implementasinya. *Mininet* adalah simulator yang mendukung protokol *OpenFlow*. *Mininet* juga bersifat realistis yang mana jika simulasi pada *Mininet* berhasil, maka dimungkinkan untuk langsung diimplementasikan pada jaringan real.

Pada saat ini, infrastruktur jaringan internet di STMIK AKAKOM masih menggunakan model tree/client server. Dengan model ini semua teknologi internet yang berbasis client server bisa terkelola dengan baik. Tetapi akan timbul permasalahan jika akan mengadopsi teknologi baru seperti cloud computing, big data, internet of things, video streaming dan voice.

Berdasar dari semua isu di atas terdapat gagasan implementasi simulator *Mininet* di lingkungan STMIK AKAKOM untuk mendukung mahasiswa dan dosen bisa melakukan penelitian dan pengujian jaringan komputer masa depan yang berbasis *Software Defined Network*.

1.2 Rumusan Masalah

Dari latar belakang masalah yang telah dikemukakan, maka dapat dirumuskan suatu masalah sebagai berikut: bagaimanakah membangun lingkungan testbed jaringan komputer masa depan berbasis *Software Defined Network* pada STMIK AKAKOM sehingga dapat membantu mahasiswa maupun dosen yang ada di lingkungan STMIK AKAKOM dapat melakukan pembuktian dari penelitiannya di bidang jaringan komputer.

1.3 Batasan Masalah

Adapun batasan masalah di dalam penelitian ini adalah:

1. Topologi jaringan yang akan diteliti adalah jaringan lokal STMIK AKAKOM.
2. Infrastruktur jaringan yang akan diteliti adalah jaringan kabel dan wireless.
3. Besarnya *bandwidth* dari operator dan data penggunaan *bandwidth* dari user disesuaikan sesuai dengan kondisi nyata.
4. Proses pengujian jaringan dilakukan dengan simulator *Mininet*.

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk membangun lingkungan *testbed* jaringan komputer masa depan berbasis *Software Defined Network* dengan menggunakan simulator *Mininet* di STMIK AKAKOM.

1.5 Manfaat Penelitian

Hasil dari penelitian ini diharapkan dapat memudahkan mahasiswa dan dosen untuk melakukan penelitian di bidang teknologi jaringan komputer masa depan yang berbasis *Software Defined Network*.

1.6 Target Luaran

Penelitian ini ditargetkan untuk dapat dipresentasikan pada Seminar Nasional Manajemen Informatika dan Komputerisasi Akuntansi 2014 (SNaMIKA 2014). Juga diharapkan hasil penelitian ini bisa digunakan sebagai bahan ajar khususnya teknologi jaringan komputer berbasis *Software Defined Network* di STMIK AKAKOM.

BAB II

TINJAUAN PUSTAKA

2.1 Teknologi jaringan komputer

Arsitektur model jaringan tradisional (model tree/client server), sudah mulai tidak bisa mendukung kebutuhan sumber daya saat ini seperti perusahaan enterprise, operator dan end user. Sebagai contoh pada model jaringan traditional, jaringan komputer berarti sebuah client mengakses ke server. Tetapi pada masa sekarang satu aplikasi bisa mengakses ke banyak database dan juga bisa ke banyak server lain secara bersamaan. Jadi perubahan pattern dari trafik adalah kebutuhan mendasar untuk user jaringan komputer saat ini [7].

Selain itu perkembangan teknologi mobile juga mendorong perubahan yang ada pada teknologi jaringan komputer. Sekarang karyawan mengakses jaringan kantor tidak hanya melalui komputer/ laptop kantor tetapi bisa juga melalui piranti pribadi seperti Ipad, Android, Iphone dan lainnya, yang dikenal dengan istilah *Bring Your Own Device (BYOD)*. Hal ini memunculkan tantangan baru pada departemen IT, sehingga data perusahaan tetap terjaga keasliannya.

Perkembangan layanan teknologi awan *cloud computing* juga mendorong perubahan pada jaringan internet. Selain itu juga adanya teknologi baru yang disebut *big data* yang berarti memerlukan kapasitas *bandwidth* yang lebar.

2.2 *Software Defined Network (SDN)*

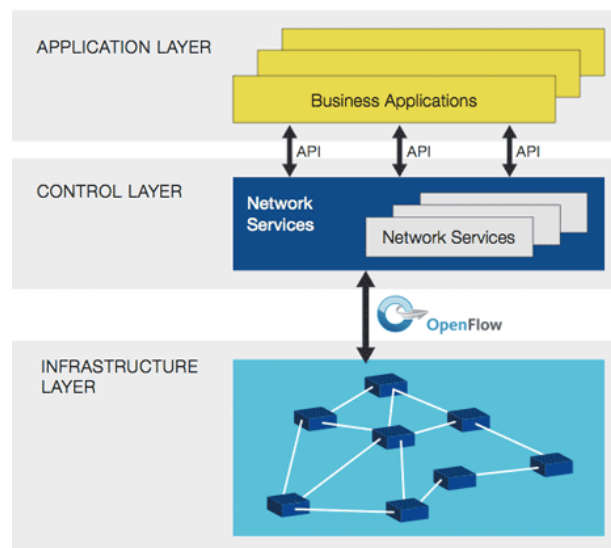
Software Defined Network (SDN) adalah arsitektur jaringan komputer model baru di mana terjadi pemisahan antara *data plane* dengan *control plane* dan bersifat programmable. Ide dari konsep di atas [7] mulai dikembangkan oleh *Open Network Foundation (ONF)*. *ONF* adalah organisasi yang mendorong promosi dan adopsi dari teknologi *Software Defined Network* melalui standarisasi yang terbuka. Pada teknologi jaringan komputer traditional (model tree/client server), aturan pengiriman data biasanya tersimpan diperangkat keras/menyatu.

Proyek *ONF* pertama kali dimulai dari 2 universitas yang berbeda yaitu UC Berkeley dan Stanford University. Dengan *SDN* administrator jaringan komputer bisa mengelola jaringannya hanya melalui abstraksi lapisan aplikasi atau

administator bisa mengkonfigurasi jaringan dengan tanpa akses ke fisik perangkat tersebut.

Protokol yang digunakan *SDN* adalah *OpenFlow*. Pada gambar 1 adalah arsitektur dari *SDN* yang tersusun dari 3 bagian yaitu lapisan aplikasi, laporan kontrol dan lapisan infrastruktur. Kemampuan dari jaringan *SDN* adalah terletak pada lapisan kontroller dimana kontroller sebagai kendali dari semua perangkat keras jaringan.

Dengan *SDN*, Perusahaan Enterprise dan Operator menjadi tidak ketergantungan kepada *vendor* dan bisa mengendalikan semua jaringan yang ada dengan sebuah aplikasi. Sehingga hal ini membuat lebih praktis dan sederhana dalam hal perancangan maupun operasional.



Gambar 1. *Software Defined Network (SDN)*

2.3 *OpenFlow*

OpenFlow adalah standard komunikasi interface yang pertama antara lapisan control dan forwarding pada arsitektur *SDN*. *OpenFlow* dikembangkan untuk mendukung peneliti bisa mengimplementasikan penelitian di bidang protokol jaringan pada lingkungan jaringan komputer kerja [5].

Dengan *OpenFlow* perubahan aturan pada switch, router fisik maupun virtual bisa dilakukan secara langsung. Saat ini teknologi jaringan komputer tersusun banyak *vendor*, sehingga untuk melakukan konfigurasi diharuskan sesuai

dengan aturan *vendor* tersebut. Atau bisa dikatakan jaringan komputer saat ini bersifat monolitik, tertutup dan seperti mainframe [5]. Protokol *OpenFlow* dibutuhkan untuk memisahkan control plane keluar dari sumber daya switch ke software kontrol yang terpusat.

OpenFlow berbasis *SDN* menawarkan banyak keuntungan seperti bisa mengelola *bandwidth* yang besar, mudah memodifikasi aturan jaringan sesuai kebutuhan, mengurangi operasional dan kompleksitas, bisa dinamis melayani permintaan aplikasi saat ini (*cloud computing, big data, internet of things,*) [7].

2.4 Network simulator (NS)

Network simulator (NS) adalah perangkat lunak yang digunakan untuk menguji performa jaringan komputer dengan tanpa perangkat yang nyata. Sehingga hal ini bisa meminimalkan biaya produksi. Di aplikasi simulator, jaringan komputer biasanya dimodelkan dengan sumber daya (*host, switch, links*), trafik dan lain sebagainya. Hal ini yang mendasari penelitian [3],[4] di mana pengujian dan evaluasi jaringan komputer bisa dilakukan pada jaringan komputer produksi.

Kemudian kemampuan dan kapasitas dari jaringan tersebut akan diamati dan dianalisis. Protokol yang didukung oleh aplikasi simulator antara lain *Wlan, Wimax, TCP, OpenFlow, SDN* dan sebagainya. Berikutnya akan dijelaskan beberapa aplikasi simulator.

2.4.1 NS3

NS3 adalah *Network simulator* gratis di bawah lisensi GNU GPLv2 yang dikhususkan untuk peneliti dan mahasiswa [2]. Tujuan dari proyek *NS3* adalah untuk membangun lingkungan simulator yang terbuka untuk penelitian di jaringan komputer. Pengembang *NS3* mendorong untuk bisa mengaplikasikan *emulator* jaringan *realtime* sehingga *NS3* bisa terkoneksi dengan jaringan nyata dan hasil dari *NS3* bisa langsung diimplementasikan

Inti dari simulator *NS3* adalah mendukung penelitian di teknologi IP maupun non *IP*. Namun kebanyakan peneliti yang menggunakan *NS3* berfokus pada teknologi *Wireless/IP* seperti *WIFI, WiMAX* atau *LTE* dari

layer 1 dan 2 dan beberapa statik atau dinamik *routing* protokol seperti *OLSR* dan *AODV* pada aplikasi berbasis IP [2].

2.4.2 *Packet Tracer*

Cisco *Packet Tracer* (PT) adalah program simulasi jaringan komputer yang dapat digunakan mahasiswa untuk melakukan percobaan jaringan komputer. PT menyediakan simulasi, visualisasi, permodelan, penilaian dan kolaborasi kemampuan dan fasilitas untuk pengajaran dan pembelajaran pada konsep teknologi jaringan komputer yang kompleks [3].

Saat ini PT mendukung teknologi dasar dari *routing* seperti *RIP*, *OSPF* dan *EIGRP* dan hal ini mendukung dari kurikulum CCNA. Meskipun PT memberikan simulasi yang realistis tetapi fitur yang ada hanya sedikit yang bisa digunakan dibanding dengan perangkat keras yang sebenarnya di Cisco IOS. Jadi PT tidak cocok untuk digunakan sebagai model jaringan komputer produksi.

2.4.3 *Mininet*

Mininet adalah emulator untuk jaringan komputer[1]. *Mininet* bisa menjalankan banyak client, switch, router maupun link dan berapa pada kernel linux. Untuk menggunakan *Mininet* cukup dengan sebuah laptop [4]. Selain itu *Mininet* menawarkan berbagai macam keunggulan seperti[1][6][8].

- *Prototyping*

Mahasiswa, Peneliti, *Network Administrator* dan dan lain-lain, dengan sumber daya tunggal seperti *Desktop PC* atau *Laptop*, bisa menjalankan *Mininet* untuk melakukan prototipe ide *SDN*.

- *Deployment*

Saat ide dari *SDN* berjalan di *Mininet*, berarti menandakan bahwa ide tersebut bisa dideploy maupun dilakukan produksi. *Mininet* memberikan fasilitas untuk mentransfer *SDN* ke sumber daya yang diinginkan.

- *Sharing*

Model yang berjalan di *Mininet* dapat dengan mudah di bagi dengan dipaket pada *Virtual Machine* maupun didistribusi ulang.

BAB III

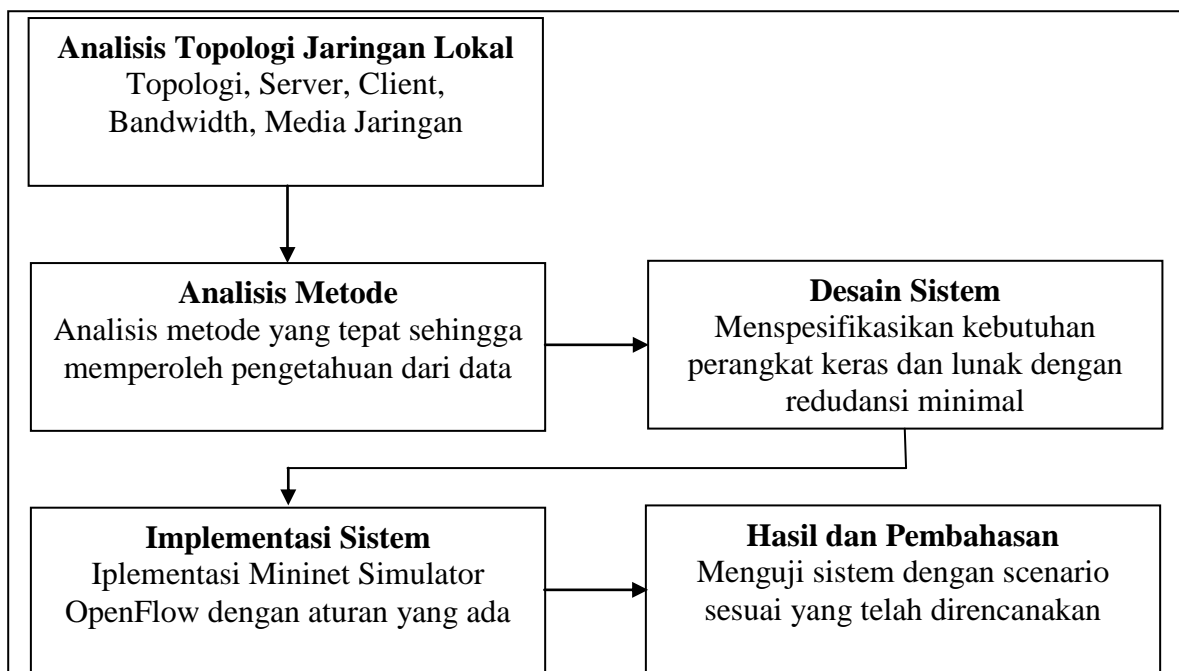
METODE PENELITIAN

3.1 Metode Penelitian

Penelitian ini diawali dengan menganalisis topologi jaringan yang ada di STMIK Akakom. Metode yang digunakan untuk menganalisis adalah wawancara langsung dengan tim Sistem dan Jaringan STMIK Akakom (SiJar), mempelajari dokumen berhubungan dengan Jaringan local dan survei langsung ke lokasi. Setelah kebutuhan data terpenuhi dilanjutkan dengan mengumpulkan data yang diperlukan sesuai pada bahan penelitian.

Langkah selanjutnya adalah mencari metode yang tepat untuk mengelola data dari tim SiJar. Hal ini dikarenakan pada topologi jaringan yang ada, ada beberapa bagian yang sama sehingga dimungkinkan hanya diwakilkan 1 sampai 2 host. Sehingga redundansi pada waktu implementasi bisa diminimalkan.

Tahap selanjutnya adalah implementasi Simulator Mininet. Proses ini diawali dengan mendeskripsikan spesifikasi kebutuhan perangkat lunak seperti server, client, media jaringan, router, switch, dilanjutkan dengan rancangan desain jaringan pada simulator Mininet. Selanjutnya sistem dikembangkan menggunakan pendekatan dari atas ke bawah.



Proses ini berakhir pada tahap pengujian sistem simulasi Mininet-OpenFlow setelah implementasi selesai dilaksanakan. Proses pengujian diawali dengan memeriksa semua koneksi yang ada. Setelah semua berfungsi dan teruji dengan baik, dilanjutkan dengan menetapkan beberapa skenario.

3.2 Infrastrukturu Jaringan STMIK Akakom

Setelah dilakukan pengumpulan data dari SiJar, dokumentasi dan studi langsung didapatkan hasil seperti dibawah ini. Pada jaringan Akakom menggunakan Internet Service Provider (ISP) lokal dengan bandwidth sebesar 13 Mbps. Bandwidth ini dirasakan sudah mencukupi dikarenakan jaringan lokal didukung dengan adanya jaringan cache dari Akamai Google 15 Mbps. Media yang digunakan untuk adalah Fiber Optik (FO) sedangkan ke Akamai Google dengan Wireless.

Untuk jaringan tulang punggung lokal saat ini sudah menggunakan infrastruktur jaringan 1 Gbps dengan media kabel UTP cat 6. Pada pengguna lokal hampir semua menggunakan infrastruktur 100 Mbps – 1 Gbps sedangkan untuk nirkabel (Akses Point) menggunakan infrastruktur dari 512 Kbps – 2 Mbps.

Tabel 3.1 Infrastruktur Jaringan Akakom

Jaringan	Bandwidth	Media
ISP	13 Mbps	Fiber Optik
Akamai Google	15 Mbps	Wireless
Tulang Punggung Lokal	1 Gbps	UTP Cat 5e,6
Pengguna Lokal	100Mbps - 1 Gbps	UTP Cat 5e
Wireless Lokal	512 Kbps – 2 Mbps	Wireless B/G/N

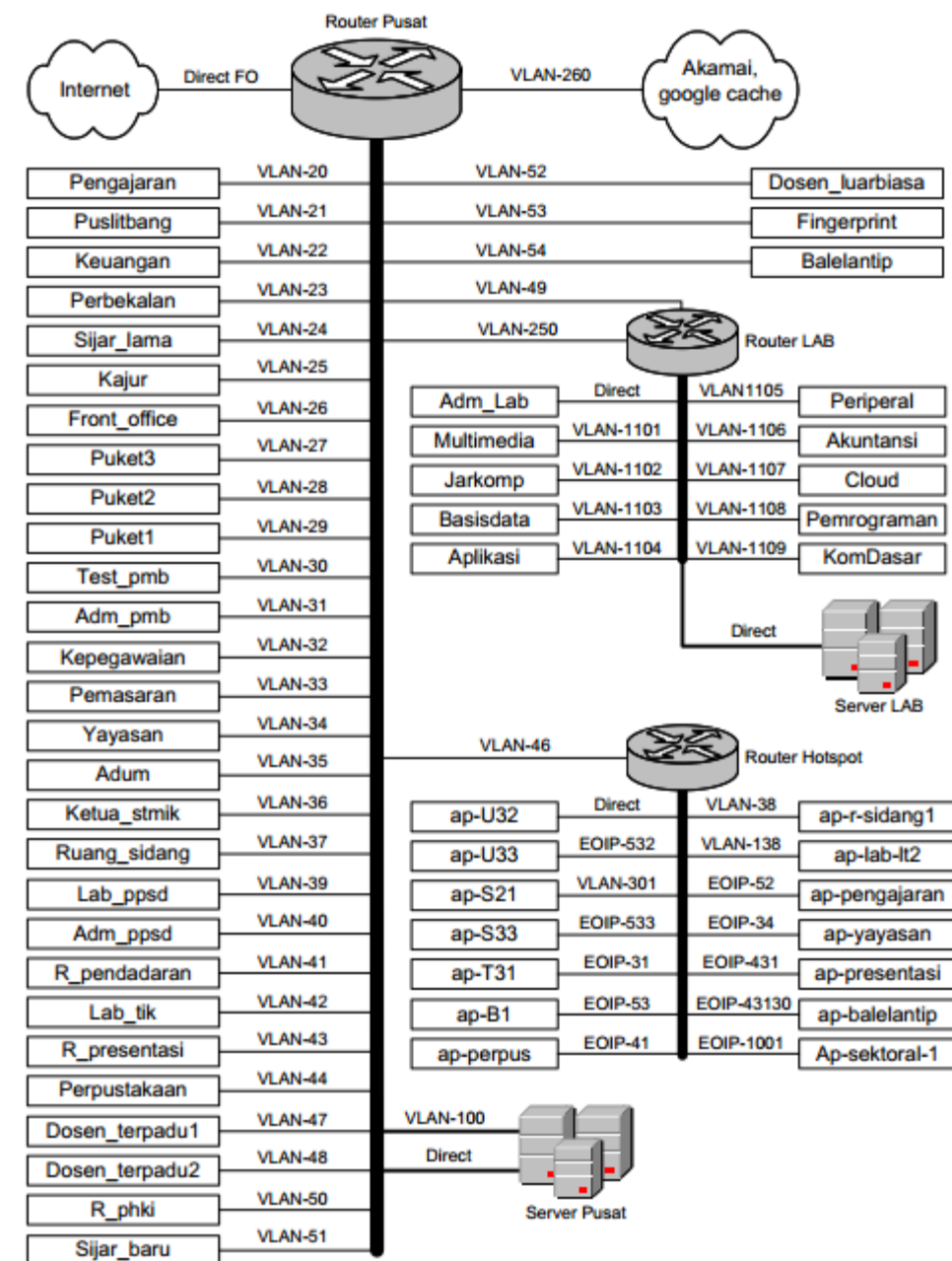
Tabel 3.2 Data Server Akakom

Server	Fungsi	Operating Sistem
SIA	Server untuk sistem akademik	Linux
Perpustakaan	Server untuk pengelolaan perpustakaan	Linux
Email	Server email untuk mahasiswa, karyawan dan dosen	Linux
Web	Server web www.akakom.ac.id	Linux
Laboratorium	Server untuk pengelolaan lab, dan media penyimpan bagi petugas lab dan mahasiswa	Linux
Hosting Virtual Machine	Server untuk praktikum cloud computing dan penyediaan hosting untuk penunjang kegiatan mahasiswa	Linux

Pada table 3.2, hampir semua server yang ada di Akakom adalah menggunakan linux. Distro yang digunakan antara lain CentOS, SlackWare dan Mandriva. Dikarenakan untuk kemudahan administrasi dan keamanan maka setiap bagian di STMIK Akakom dapat dipastikan mempunyai server masing-masing.

Untuk saat ini server yang ada tidak hanya yang berupa fisik tetapi sudah ada yang berupa virtual machine. Contoh server yang menggunakan virtual machine adalah server cloud computing yang ada di laboratorium.

Infrastruktur jaringan Akakom secara logika seperti yang tertera pada di bawah tersusun dari banyaknya Vlan. Alasan penggunaan vlan adalah kemudian administrasi dan keamanan. Seperti tersebut digambar bahwa jaringan Akakom mempunyai 1 Router utama, 1 router Laboratorium dan 1 router AP. Merk yang digunakan adalah Cisco, Mikrotik, TP Link. Jumlah vlan yang ada sekitar 40 vlan.

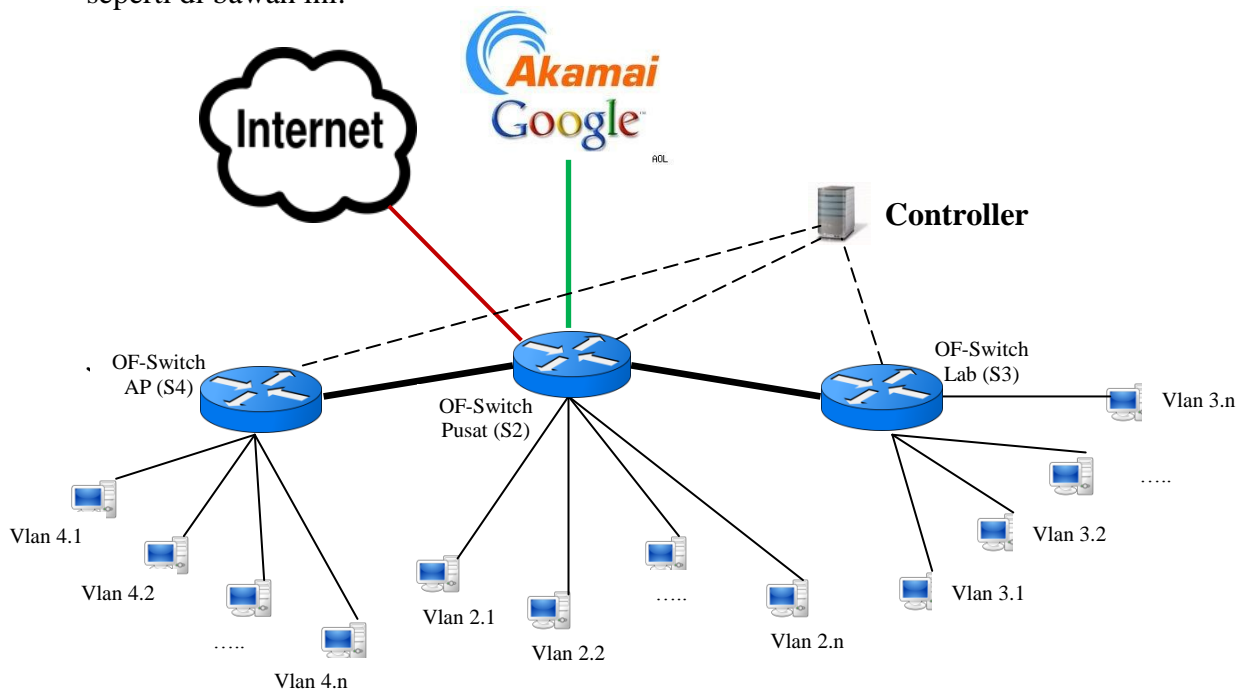


Gambar 3.1 Topologi Jaringan Logika Akakom

3.3 Analisis dan Desain

Setelah kebutuhan akan data terpenuhi, dilanjutkan dengan analisis dan design untuk memetakan kebutuhan apa saja yang akan diperlukan untuk melakukan simulasi di *Mininet*. Hal yang diamati adalah Topologi jaringan secara keluruhan, perangkat keras, pembagian ip address dan rules yang berlaku pada jaringan komputer STMIK AKAKOM.

Dikarenakan pada jaringan Akakom menggunakan lebih dari 40 vlan, maka pada proses analisis dan desain diputuskan untuk membuat vlan hanya beberapa. Hal ini dikarenakan mengurangi adanya duplikasi. Detail dari desain adalah seperti di bawah ini.



Gambar 3.2. Desain topologi simulator

Pada bagian ini, hampir semua bagian bisa terwakili mulai dari ISP, Akamai Google dan vlan hanya saja untuk vlan sebanyak 4 vlan per switch. Pada semua router diganti menggunakan OF-Switch (OpenFlow Switch) dengan 1 controller. Controller ini berfungsi sebagai control plane dari data yang ada di switch. Bedanya dengan jaringan komputer saat ini adalah control plane biasanya berada pada alat yang sama dengan data plane (embedded). Detail link, bandwidth dan media seperti tabel di bawah ini.

Tabel 3.3 Link, Bandwidth dan Media

Link	Bandwidth	Media
ISP – OF-Switch Pusat (S2)	13 Mbps	Fiber Optic
Akakai - OF-Switch Pusat (S2)	15 Mbps	Wireless
OF-Switch Pusat (S2) - OF-Switch Lab (S3)	100 Mbps	UTP
OF-Switch (S2) - OF-Switch AP(S4)	100 Mbps	UTP
OF-Switch Pusat (S2) – Vlan2.1	2 Mbps	UTP
OF-Switch Pusat (S2) – Vlan2.2	2 Mbps	UTP
OF-Switch Pusat (S2) – Vlan2.N	2 Mbps	UTP
OF-Switch Lab (S3) – Vlan3.1	1 Mbps	UTP
OF-Switch Lab (S3) – Vlan3.2	1 Mbps	UTP
OF-Switch Lab (S3) – Vlan3.N	1 Mbps	UTP
OF-Switch AP (S4) – Vlan4.1	1 Mbps	UTP
OF-Switch AP (S4) – Vlan4.2	1 Mbps	UTP
OF-Switch AP (S4) – Vlan4.N	1 Mbps	UTP

Tabel 3.4 Kebutuhan sistem

Nama	Fungsi
Ubuntu 14.04	Server Controller OpenFlow
Open VSwitch 2.02	Controller
Ping	Tool jaringan
Iperf	Tool bandwidth
Jperf	Tool graph bandwidth

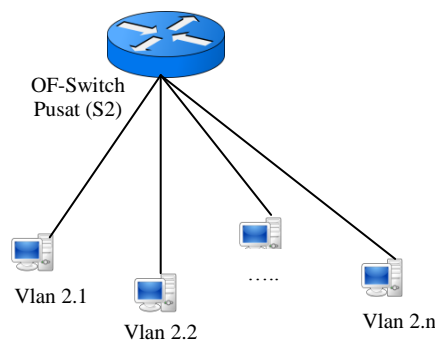
Tabel 3.4 Kebutuhan IP

Host	IP
Vlan 2.1 – Vlan 2.4	10.0.0.1/24 – 10.0.0.4/24
Vlan 3.1 – Vlan 3.4	10.0.0.5/24 – 10.0.0.8/24
Vlan 4.1 – Vlan 4.4	10.0.0.9/24 – 10.0.0.12/24
ISP	10.0.0.13/24
Akamai Google	10.0.0.14/24
Controller	127.0.0.1/24

3.4 Implementasi

Proses implementasi diawali dengan menentukan skenario yang nantinya akan dianalisa untuk memperoleh suatu pengetahuan. Pada proses ini dilakukan dari proses bawah ke atas sehingga memudahkan dalam penelusuran kesalahan.

Proses pertama adalah implementasi router utama (S2) dengan 4 vlan. Perintah yang digunakan untuk membuat topologi seperti di bawah adalah sebagai berikut.



Add hosts and switches

```
h1 = self.addHost( 'h1' ) #membuat host vlan 2.1
h2 = self.addHost( 'h2' ) #membuat host vlan 2.2
h3 = self.addHost( 'h3' ) #membuat host vlan 2.3
h4 = self.addHost( 'h4' ) #membuat host vlan 2.4
s2 = self.addSwitch( 's2' ) #membuat OpenFlow switch Pusat (S2)
```

Add links

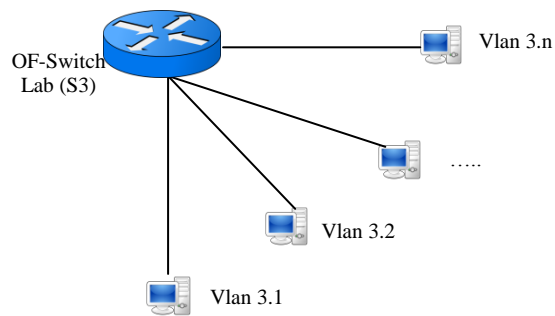
```
self.addLink( h1, s2, bw = 1000, delay = '1ms' )
#link dari host 2.1 ke S2 bandwidth 1 Gbps dan Delay = 1 ms

self.addLink( h2, s2, bw = 1000, delay = '1ms' )
#link dari host 2.2 ke S2 bandwidth 1 Gbps dan Delay = 1 ms

self.addLink( h3, s2, bw = 100, delay = '1ms' )
#link dari host 2.3 ke S2 bandwidth 100 Mbps dan Delay = 1 ms

self.addLink( h4, s2, bw = 100, delay = '1ms' )
#link dari host 2.4 ke S2 bandwidth 100 Mbps dan Delay = 1 ms
```

Proses kedua adalah implementasi router kedua (S3) dengan 4 vlan. Perintah yang digunakan untuk membuat topologi seperti di bawah adalah sebagai berikut.



```
# Add hosts and switches
h11 = self.addHost( 'h11' ) #membuat host vlan 3.1
h12 = self.addHost( 'h12' ) #membuat host vlan 3.2
h13 = self.addHost( 'h13' ) #membuat host vlan 3.3
h14 = self.addHost( 'h14' ) #membuat host vlan 3.4
s3 = self.addSwitch( 's3' ) #membuat OpenFlow switch Lab (S3)

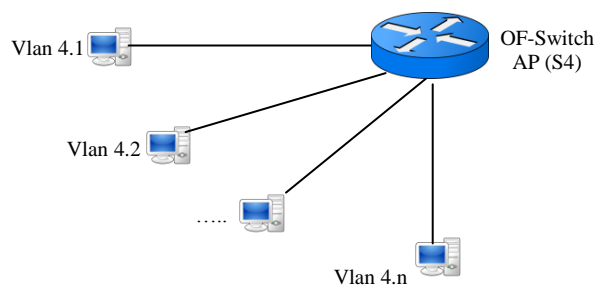
# Add links
self.addLink( h11, s3, bw = 100, delay = '1ms' )
#link dari host 3.1 ke S3 bandwidth 100 Mbps dan Delay = 1 ms

self.addLink( h12, s3, bw = 100, delay = '1ms' )
#link dari host 3.2 ke S3 bandwidth 100 Mbps dan Delay = 1 ms

self.addLink( h13, s3, bw = 100, delay = '1ms' )
#link dari host 3.3 ke S3 bandwidth 100 Mbps dan Delay = 1 ms

self.addLink( h14, s3, bw = 100, delay = '1ms' )
#link dari host 3.4 ke S3 bandwidth 100 Mbps dan Delay = 1 ms
```

Proses ketiga adalah implementasi router ketiga (S4) dengan 4 vlan. Perintah yang digunakan untuk membuat topologi seperti di bawah adalah sebagai berikut.



```
# Add hosts and switches
h21 = self.addHost( 'h21' ) #membuat host vlan 4.1
h22 = self.addHost( 'h22' ) #membuat host vlan 4.2
h23 = self.addHost( 'h23' ) #membuat host vlan 4.3
h24 = self.addHost( 'h24' ) #membuat host vlan 4.4
s4 = self.addSwitch( 's4' ) #membuat OpenFlow switch AP (S4)
```

```
# Add links
self.addLink( h21, s4, bw = 100, delay = '1ms' )
#link dari host 4.1 ke S4 bandwidth 100 Mbps dan Delay = 1 ms

self.addLink( h22, s4, bw = 100, delay = '1ms' )
#link dari host 4.2 ke S4 bandwidth 100 Mbps dan Delay = 1 ms

self.addLink( h23, s4, bw = 100, delay = '1ms' )
#link dari host 4.3 ke S4 bandwidth 100 Mbps dan Delay = 1 ms

self.addLink( h24, s4, bw = 100, delay = '1ms' )
#link dari host 4.4 ke S4 bandwidth 100 Mbps dan Delay = 1 ms
```

Proses keempat adalah implementasi router pusat (S2) dengan 4 vlan dan link ke router S3 dan router S4. Perintah yang digunakan untuk membuat topologi seperti di bawah adalah sebagai berikut.

```
# Add hosts and switches
h1 = self.addHost( 'h1' ) #membuat host vlan 1.1
h2 = self.addHost( 'h2' ) #membuat host vlan 1.2
h3 = self.addHost( 'h3' ) #membuat host vlan 1.3
h4 = self.addHost( 'h4' ) #membuat host vlan 1.4
h100 = self.addHost( 'h100' ) #Internet Service Provider
h101 = self.addHost( 'h101' ) #Akamai Google
s2 = self.addSwitch( 's2' ) #membuat OpenFlow switch Pusat (S2)

# Add links
self.addLink( h100, s2, bw = 13, delay = '5ms' )
#link ISP ke S2 bandwidth 13 Mbps dan Delay = 5 ms

self.addLink( h101, s2, bw = 15, delay = '10ms' )
#link Akamai Google ke S2 bandwidth 15 Mbps dan Delay = 10 ms

self.addLink( h1, s2, bw = 1000, delay = '1ms' )
#link dari host 1.1 ke S2 bandwidth 1 Gbps dan Delay = 1 ms

self.addLink( h2, s2, bw = 1000, delay = '1ms' )
#link dari host 2.2 ke S2 bandwidth 1 Gbps dan Delay = 1 ms

self.addLink( h3, s2, bw = 100, delay = '1ms' )
#link dari host 1.3 ke S2 bandwidth 100 Mbps dan Delay = 1 ms

self.addLink( h4, s2, bw = 100, delay = '1ms' )
#link dari host 1.4 ke S2 bandwidth 100 Mbps dan Delay = 1 ms

self.addLink( s3, s2, bw = 100, delay = '3ms' )
#link dari S3 ke S2 bandwidth 100 Mbps dan Delay = 3 ms

self.addLink( s4, s2, bw = 100, delay = '3ms' )
#link dari S4 ke S2 bandwidth 100 Mbps dan Delay = 3 ms
```

Setelah proses implementasi topologi, perlu dilakukan proses add-flow pada switch OpenFlow

```
#mengkoneksi link in-out pada OF-Switch S2 (normal)
```

```
mininet>sh ovs-ofctl add-flow s2 action=normal #OF-Switch S2  
mininet>sh ovs-ofctl add-flow s3 action=normal #OF-Switch S3  
mininet>sh ovs-ofctl add-flow s4 action=normal #OF-Switch S4
```

3.5 Simulasi

Pada bagian simulasi, dilakukan 4 skenario percobaan jaringan sesuai dengan kondisi jaringan Akakom. Detail dari simulasi seperti di bawah ini.

1. Semua Vlan (h1 sampai h21 terkoneksi dan bisa melakukan ping ke server ISP (h100) dan server Akamai Google (h101).
2. Pengujian bandwidth dari ISP maupun Akamai.
3. Pengujian bandwidth dari ISP dan Akamai Google di client dan berdasar aturan yang berlaku di jaringan Akakom.
4. Jaringan Laboratorium tidak terkoneksi ke ISP dan Akamai Google.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Penelitian

Mininet adalah simulator jaringan yang mendukung implementasi protokol OpenFlow. Pada mininet didukung berbagai macam teknologi switch antara lain Openvswitch, POX, Floodlight. Keluaran yang didapat dari simulasi ini adalah dengan Mininet kita bisa mendesain sebuah jaringan masa depan dimana memungkinkan untuk mengimplementasikan protocol OpenFlow. Pada bagian ini akan dijelaskan hasil dari 4 poin simulasi pada bab 3.

4.1.1 Semua vlan terhubung dengan ISP maupun Akamai Google

Untuk melakukan simulasi ini, pertama yang dilakukan adalah membuat script topologi. Script berisi sejumlah host dan switch beserta besarnya bandwidth maupun delay. Untuk menjalankan script caranya dengan :

```
mininet@mininet-vm:~$ sudo mn --custom akakomOK-real-topo.py --topo mytopo --controller=remote
,ip=127.0.0.1 --mac --link tc
```

Membuat topologi dari file akakomOK-real-topo.py dengan controller remote dan mac address sesuai dengan ip address.

Jika tidak ada kesalahan dalam script maka akan tertampil seperti berikut ini. Kemudian dilakukan pengujian jaringan dengan tool ping.

```
mininet@mininet-vm:~$ sudo mn --custom akakomOK-real-topo.py --topo mytopo --link tc --mac --controller=remote,ip=127.0.0.1:6633
*** Creating network
*** Adding controller
Unable to contact the remote controller at 127.0.0.1:6633:6633
*** Adding hosts:
h1 h2 h3 h4 h11 h12 h13 h14 h21 h22 h23 h24 h100 h101
*** Adding switches:
s2 s3 s4
*** Adding links:
(1000.00Mbit 1ms delay) (1000.00Mbit 1ms delay) (h1, s2) (1000.00Mbit 1ms delay) (1000.00Mbit 1ms delay) (h2, s2) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (h3, s2) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (h4, s2) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (h11, s3) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (h12, s3) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (h13, s3) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (h14, s3) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (h21, s4) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (h22, s4) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (h23, s4) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (h24, s4) (13.00Mbit 5ms delay) (13.00Mbit 5ms delay) (h100, s2) (15.00Mbit 10ms delay) (15.00Mbit 10ms delay) (h101, s2) (100.00Mbit 3ms delay) (100.00Mbit 3ms delay) (s2, s3) (100.00Mbit 3ms delay) (100.00Mbit 3ms delay) (s2, s4)
*** Configuring hosts
h1 h2 h3 h4 h11 h12 h13 h14 h21 h22 h23 h24 h100 h101
*** Starting controller
*** Starting 3 switches
```

s2 (13.00Mbit 5ms delay) (15.00Mbit 10ms delay) (1000.00Mbit 1ms delay) (1000.00Mbit 1ms delay) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (100.00Mbit 3ms delay) (100.00Mbit 3ms delay) s3 (100.00Mbit 3ms delay) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) s4 (100.00Mbit 3ms delay) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay) (100.00Mbit 1ms delay)

*** Starting CLI:
mininet> pingall



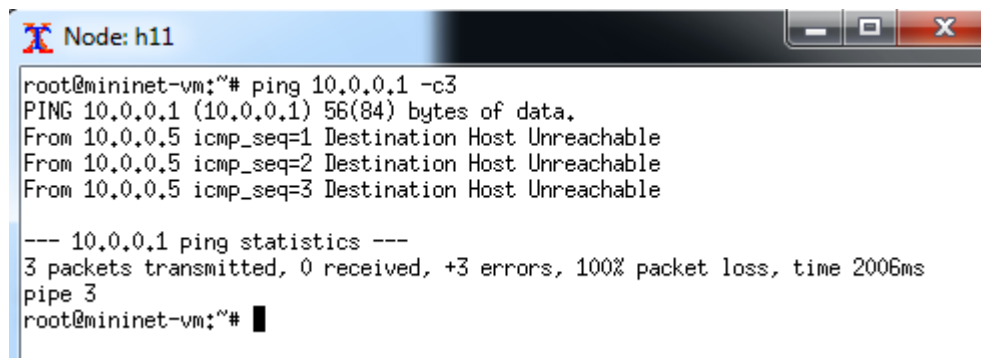
```

Node: h1
root@mininet-vm:~# ping 10.0.0.2 -c3
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
From 10.0.0.1 icmp_seq=1 Destination Host Unreachable
From 10.0.0.1 icmp_seq=2 Destination Host Unreachable
From 10.0.0.1 icmp_seq=3 Destination Host Unreachable

--- 10.0.0.2 ping statistics ---
3 packets transmitted, 0 received, +3 errors, 100% packet loss, time 1999ms
pipe 3
root@mininet-vm:~#

```

Gambar 4.1 h1 ping h2



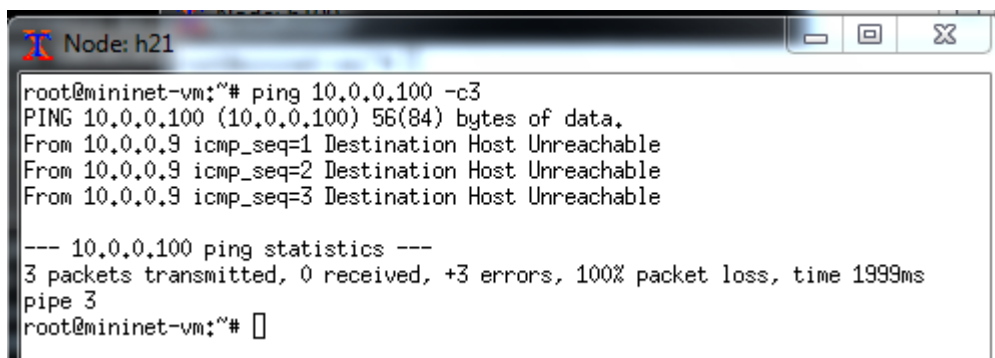
```

Node: h11
root@mininet-vm:~# ping 10.0.0.1 -c3
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
From 10.0.0.5 icmp_seq=1 Destination Host Unreachable
From 10.0.0.5 icmp_seq=2 Destination Host Unreachable
From 10.0.0.5 icmp_seq=3 Destination Host Unreachable

--- 10.0.0.1 ping statistics ---
3 packets transmitted, 0 received, +3 errors, 100% packet loss, time 2006ms
pipe 3
root@mininet-vm:~#

```

Gambar 4.2 h11 ping h1



```

Node: h21
root@mininet-vm:~# ping 10.0.0.100 -c3
PING 10.0.0.100 (10.0.0.100) 56(84) bytes of data.
From 10.0.0.9 icmp_seq=1 Destination Host Unreachable
From 10.0.0.9 icmp_seq=2 Destination Host Unreachable
From 10.0.0.9 icmp_seq=3 Destination Host Unreachable

--- 10.0.0.100 ping statistics ---
3 packets transmitted, 0 received, +3 errors, 100% packet loss, time 1999ms
pipe 3
root@mininet-vm:~#

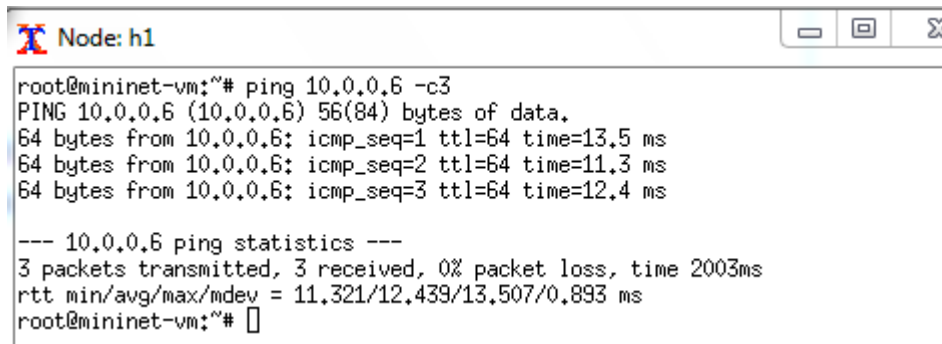
```

Gambar 4.3 h21 ping ISP

Dilakukan penambahan script *add-flow* pada S2, S3 dan S4 supaya bisa melakukan forwarding data. Perintah seperti di bawah ini.

```
mininet> sh ovs-ofctl add-flow s2 action=normal
```

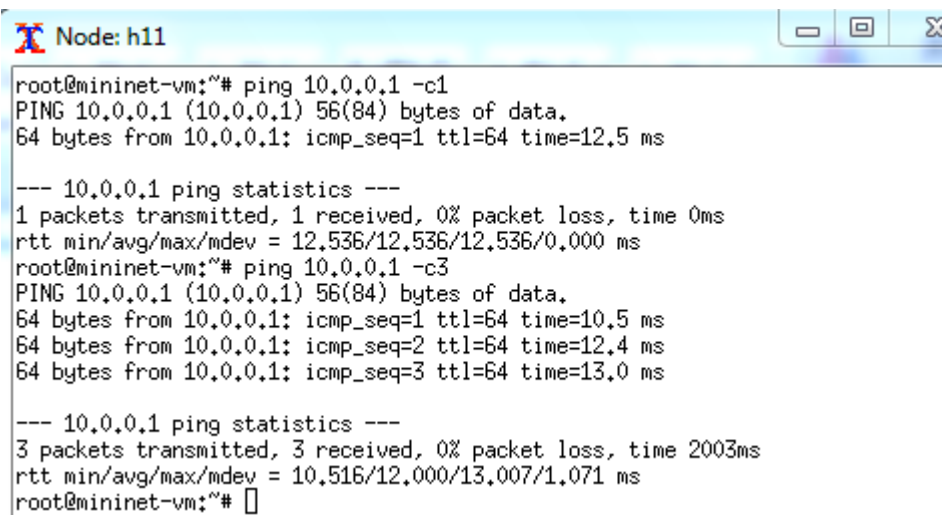
```
mininet> sh ovs-ofctl add-flow s3 action=normal
mininet> sh ovs-ofctl add-flow s4 action=normal
```



```
Node: h1
root@mininet-vm:~# ping 10.0.0.6 -c3
PING 10.0.0.6 (10.0.0.6) 56(84) bytes of data.
64 bytes from 10.0.0.6: icmp_seq=1 ttl=64 time=13.5 ms
64 bytes from 10.0.0.6: icmp_seq=2 ttl=64 time=11.3 ms
64 bytes from 10.0.0.6: icmp_seq=3 ttl=64 time=12.4 ms

--- 10.0.0.6 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 11.321/12.439/13.507/0.893 ms
root@mininet-vm:~#
```

Gambar 4.4 h1 ping h13

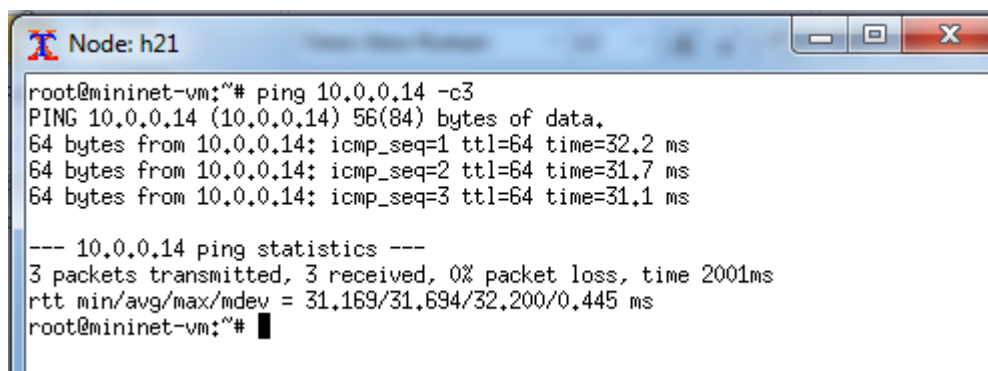


```
Node: h11
root@mininet-vm:~# ping 10.0.0.1 -c1
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=12.5 ms

--- 10.0.0.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 12.536/12.536/12.536/0.000 ms
root@mininet-vm:~# ping 10.0.0.1 -c3
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=10.5 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=12.4 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=13.0 ms

--- 10.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 10.516/12.000/13.007/1.071 ms
root@mininet-vm:~#
```

Gambar 4.5 h11 ping h1



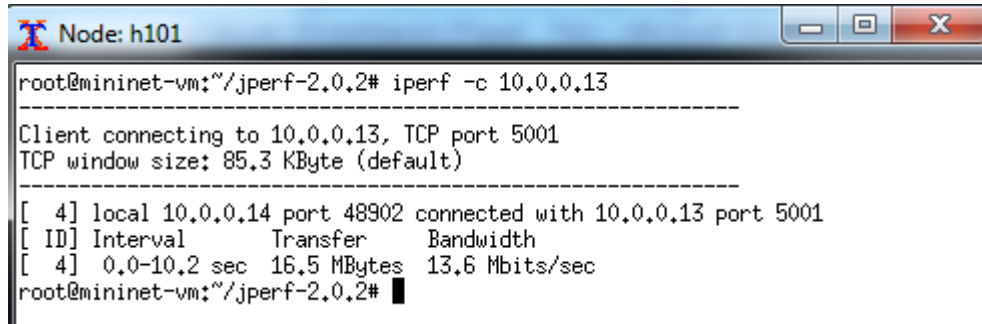
```
Node: h21
root@mininet-vm:~# ping 10.0.0.14 -c3
PING 10.0.0.14 (10.0.0.14) 56(84) bytes of data.
64 bytes from 10.0.0.14: icmp_seq=1 ttl=64 time=32.2 ms
64 bytes from 10.0.0.14: icmp_seq=2 ttl=64 time=31.7 ms
64 bytes from 10.0.0.14: icmp_seq=3 ttl=64 time=31.1 ms

--- 10.0.0.14 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 31.169/31.694/32.200/0.445 ms
root@mininet-vm:~#
```

Gambar 4.6 h21 ping Akamai Google

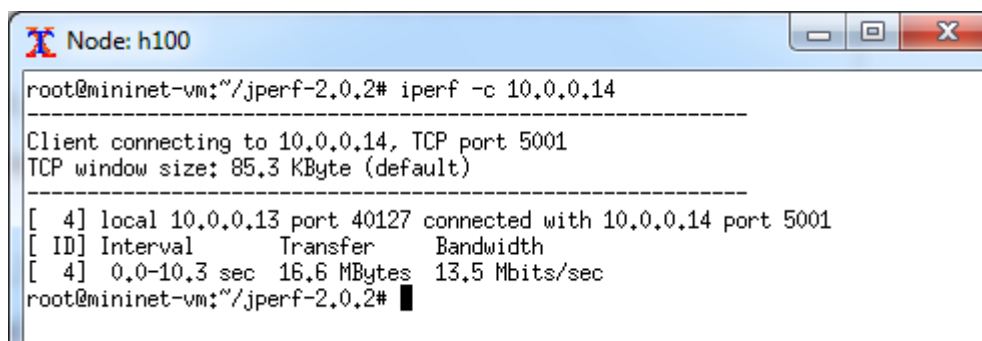
4.1.2 Pengujian bandwidth dari ISP dan Akamai Google

Pada bagian ini, pengujian bandwidth dilakukan menggunakan tool iperf. Untuk menggunakannya iperf dijalankan pada 2 sisi yaitu client dan server. Untuk mengetahui bandwidth dari ISP maka pada sisi ISP dijalankan perintah iperf -s Detail hasil seperti di bawah ini.



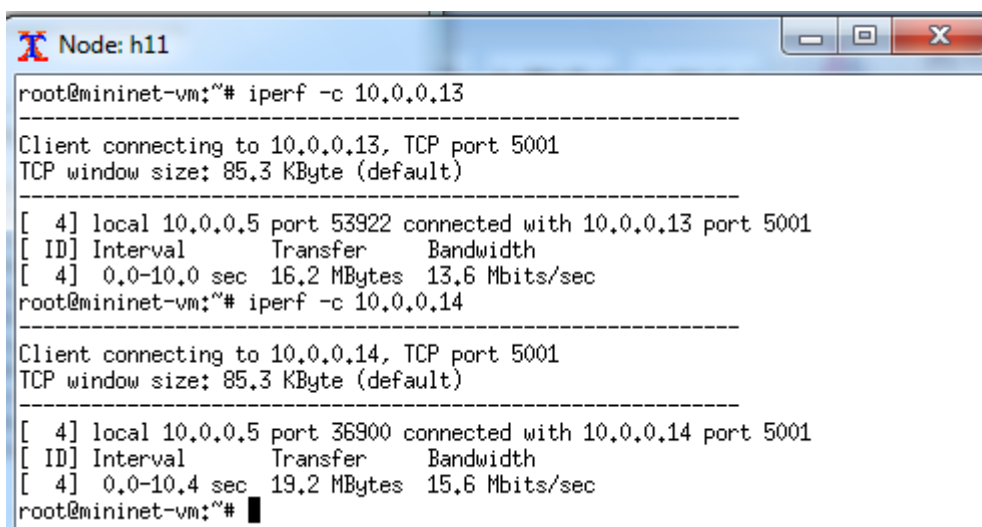
```
Node: h101
root@mininet-vm:~/jperf-2.0.2# iperf -c 10.0.0.13
-----
Client connecting to 10.0.0.13, TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[  4] local 10.0.0.14 port 48902 connected with 10.0.0.13 port 5001
[ ID] Interval      Transfer    Bandwidth
[  4] 0.0-10.2 sec  16.5 MBytes  13.6 Mbits/sec
root@mininet-vm:~/jperf-2.0.2#
```

Gambar 4.7 Bandwidth ISP



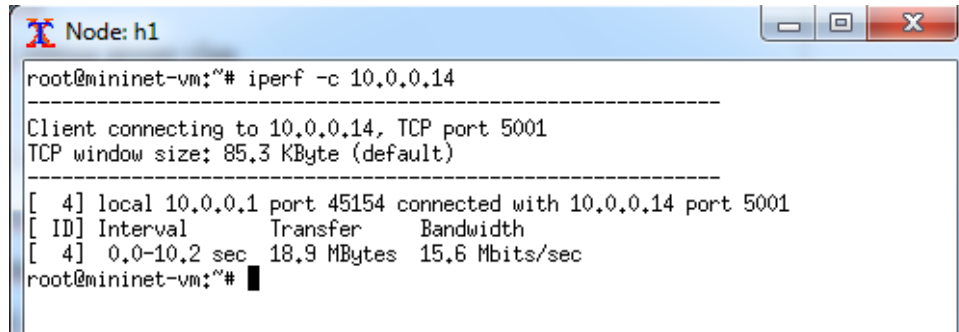
```
Node: h100
root@mininet-vm:~/jperf-2.0.2# iperf -c 10.0.0.14
-----
Client connecting to 10.0.0.14, TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[  4] local 10.0.0.13 port 40127 connected with 10.0.0.14 port 5001
[ ID] Interval      Transfer    Bandwidth
[  4] 0.0-10.3 sec  16.6 MBytes  13.5 Mbits/sec
root@mininet-vm:~/jperf-2.0.2#
```

Gambar 4.8 Bandwidth Akamai Google



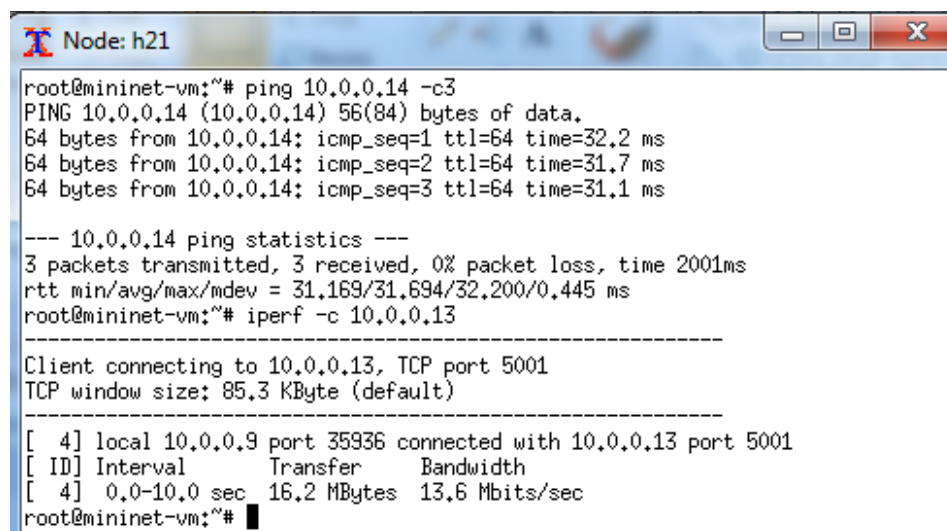
```
Node: h11
root@mininet-vm:~# iperf -c 10.0.0.13
-----
Client connecting to 10.0.0.13, TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[  4] local 10.0.0.5 port 53922 connected with 10.0.0.13 port 5001
[ ID] Interval      Transfer    Bandwidth
[  4] 0.0-10.0 sec  16.2 MBytes  13.6 Mbits/sec
root@mininet-vm:~# iperf -c 10.0.0.14
-----
Client connecting to 10.0.0.14, TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[  4] local 10.0.0.5 port 36900 connected with 10.0.0.14 port 5001
[ ID] Interval      Transfer    Bandwidth
[  4] 0.0-10.4 sec  19.2 MBytes  15.6 Mbits/sec
root@mininet-vm:~#
```

Gambar 4.9 Bandwidth h11 ke ISP dan Akamai Google



```
Node: h1
root@mininet-vm:~# iperf -c 10.0.0.14
-----
Client connecting to 10.0.0.14, TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[  4] local 10.0.0.1 port 45154 connected with 10.0.0.14 port 5001
[ ID] Interval      Transfer    Bandwidth
[  4]  0.0-10.2 sec  18.9 MBytes  15.6 Mbits/sec
root@mininet-vm:~#
```

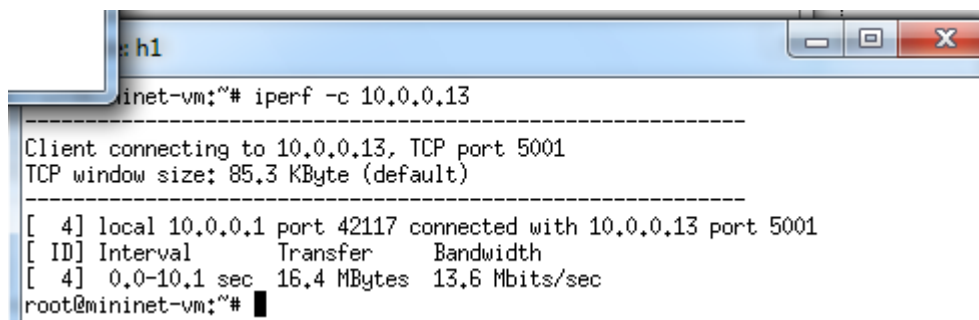
Gambar 4.10 Bandwidth h1 ke Akamai Google



```
Node: h21
root@mininet-vm:~# ping 10.0.0.14 -c3
PING 10.0.0.14 (10.0.0.14) 56(84) bytes of data:
64 bytes from 10.0.0.14: icmp_seq=1 ttl=64 time=32.2 ms
64 bytes from 10.0.0.14: icmp_seq=2 ttl=64 time=31.7 ms
64 bytes from 10.0.0.14: icmp_seq=3 ttl=64 time=31.1 ms

--- 10.0.0.14 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 31.169/31.694/32.200/0.445 ms
root@mininet-vm:~# iperf -c 10.0.0.13
-----
Client connecting to 10.0.0.13, TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[  4] local 10.0.0.9 port 35936 connected with 10.0.0.13 port 5001
[ ID] Interval      Transfer    Bandwidth
[  4]  0.0-10.0 sec  16.2 MBytes  13.6 Mbits/sec
root@mininet-vm:~#
```

Gambar 4.11 Bandwidth h21 ke ISP

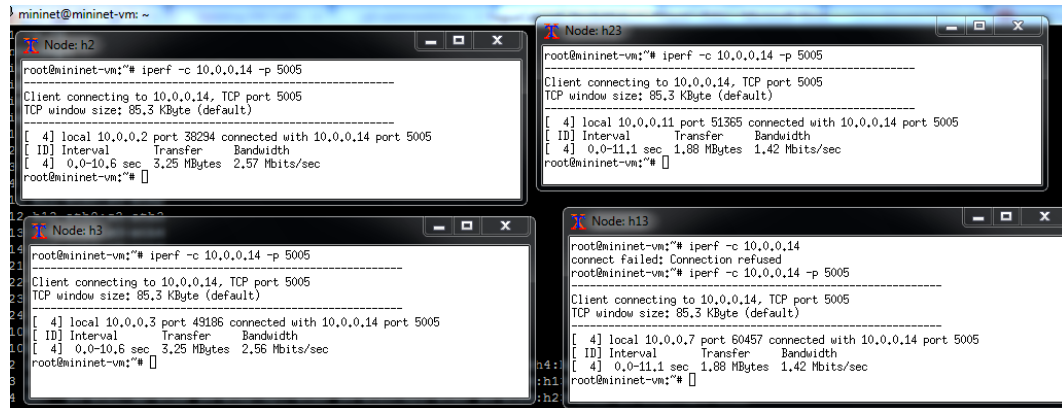


```
h1
mininet-vm:~# iperf -c 10.0.0.13
-----
Client connecting to 10.0.0.13, TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[  4] local 10.0.0.1 port 42117 connected with 10.0.0.13 port 5001
[ ID] Interval      Transfer    Bandwidth
[  4]  0.0-10.1 sec  16.4 MBytes  13.6 Mbits/sec
root@mininet-vm:~#
```

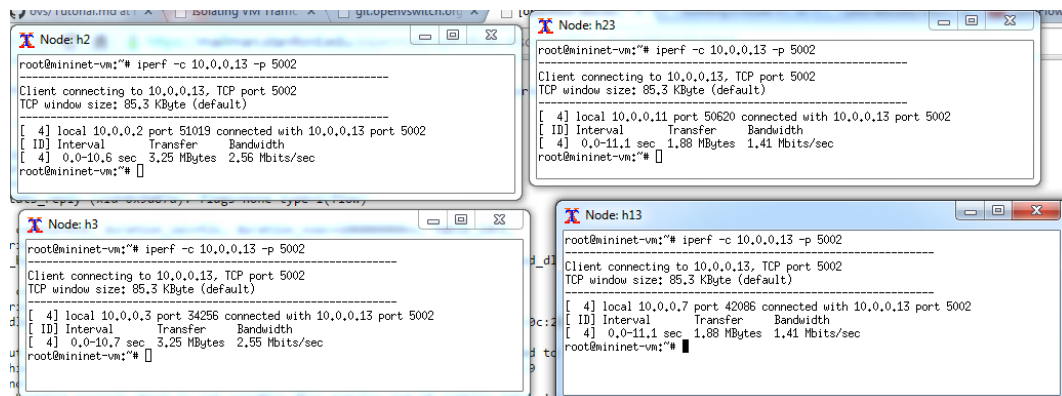
Gambar 4.12 Bandwidth h1 ke ISP

4.1.3 Pengujian bandwidth di client berdasar aturan jaringan Akakom

Pada bagian ini, pengujian bandwidth di sisi client dengan penerapan aturan pembagian bandwidth sesuai jaringan Akakom.



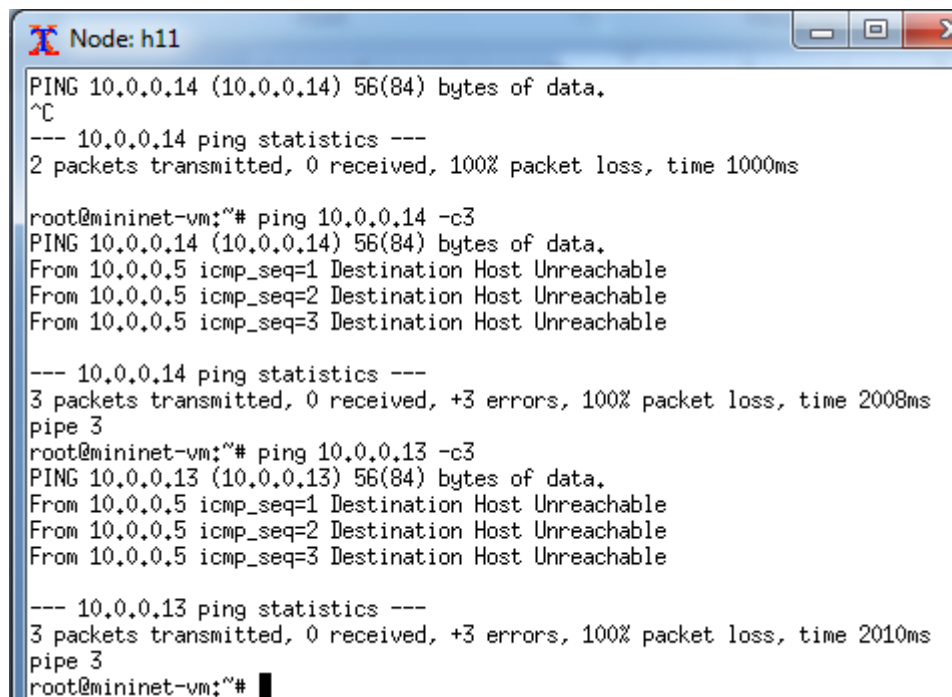
Gambar 4.13 Bandwidth di Client ke Akamai Google



Gambar 4.14 Bandwidth di Client ke ISP

4.1.4 Jaringan Laboratorium tidak terhubung ke ISP dan Akamai Google

Pada bagian ini, yang dilakukan adalah semua packet data yang berasal dari S3 menuju ISP dan Akamai Google secara otomatis di DROP.



```
Node: h11
PING 10.0.0.14 (10.0.0.14) 56(84) bytes of data.
^C
--- 10.0.0.14 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1000ms

root@mininet-vm:~# ping 10.0.0.14 -c3
PING 10.0.0.14 (10.0.0.14) 56(84) bytes of data.
From 10.0.0.5 icmp_seq=1 Destination Host Unreachable
From 10.0.0.5 icmp_seq=2 Destination Host Unreachable
From 10.0.0.5 icmp_seq=3 Destination Host Unreachable

--- 10.0.0.14 ping statistics ---
3 packets transmitted, 0 received, +3 errors, 100% packet loss, time 2008ms
pipe 3
root@mininet-vm:~# ping 10.0.0.13 -c3
PING 10.0.0.13 (10.0.0.13) 56(84) bytes of data.
From 10.0.0.5 icmp_seq=1 Destination Host Unreachable
From 10.0.0.5 icmp_seq=2 Destination Host Unreachable
From 10.0.0.5 icmp_seq=3 Destination Host Unreachable

--- 10.0.0.13 ping statistics ---
3 packets transmitted, 0 received, +3 errors, 100% packet loss, time 2010ms
pipe 3
root@mininet-vm:~#
```

Gambar 4.15 Host Lab ke ISP dan Akamai Google

4.2 Pembahasan Hasil

Di mininet pada waktu menggenerate topologi, jika hanya menuliskan – controller=remote atau controller=none maka secara otomatis tidak ada aturan di dalam switch sehingga meskipun secara logika terhubung host tidak bisa saling terhubung. Seperti pada gambar 4.1, 4.2 dan 4.3.

Pada simulasi berikutnya saat controller di tambah sintak ovs-ofctl add-flow (nama switch) action = (pilihan) maka switch mempunyai kemampuan memforward data ke setiap interfacenya. Detail hasil seperti pada gambar 4.4, 4.5 dan 4.6 yang kesemuanya menunjukkan ping berhasil.

```
mininet> sh ovs-ofctl add-flow s2 action=normal
mininet> sh ovs-ofctl add-flow s3 action=normal
mininet> sh ovs-ofctl add-flow s4 action=normal
```

Seperti pada gambar 4.7 dan 4.8 pengujian bandwidth ISP dan Akamai Google, kenyataanya bandwidth Akamai Google hanya 13.8 Gbps. Hal ini karena pengujian di Mininet hanya bisa dilakukan antara ISP dan Akamai Google. Oleh karena thoughput jaringan adalah nilai minimal dari komponen jaringan tersebut, sehingga bandwidth di Akamai Google tertampil 13.6 Gbps.

$$Throughput = \text{Min}\{Rate_{Server}, Rate_{Client}, Rate_{Koneksi}\}$$

Bandwidth yang diterima client dari ISP maupun Akamai Google adalah sesuai dengan data jaringan Akakom yaitu 13 Gbps (ISP) dan 15 Gbps (Akamai Google) seperti pada gambar 4.9, 4.10, 4.11, 4.12.

Pada hasil gambar 4.13 dan 4.14 bandwidth yang diterima tiap host menjadi kecil karena pengaruh dari penerapan aturan yang berlaku pada jaringan Akakom. Aturan pembagian bandwidth tiap host seperti di bawah ini. Meskipun infrastruktur yang digunakan sudah mendukung 100 Mbps ataupun 1 Gbps, jikalau ada pembatasan bandwidth maka secara otomatis akan mengikuti aturan tersebut.

```
self.addLink( h100, s2, bw = 13, delay = '5ms' )
self.addLink( h101, s2, bw = 15, delay = '10ms' )
self.addLink( h1, s2, bw = 2, delay = '1ms' ) #maksimal bw = 2 Mbps
self.addLink( h2, s2, bw = 2, delay = '1ms' )
self.addLink( h3, s2, bw = 2, delay = '1ms' ) #maksimal bw = 2 Mbps
self.addLink( h4, s2, bw = 1, delay = '1ms' )
self.addLink( s3, s2, bw = 100, delay = '3ms' )
self.addLink( h11, s3, bw = 1, delay = '1ms' )
self.addLink( h12, s3, bw = 1, delay = '1ms' )
self.addLink( h13, s3, bw = 1, delay = '1ms' )
self.addLink( h14, s3, bw = 1, delay = '1ms' )
self.addLink( s4, s2, bw = 100, delay = '3ms' )
self.addLink( h21, s4, bw = 1, delay = '1ms' )
self.addLink( h22, s4, bw = 1, delay = '1ms' )
self.addLink( h23, s4, bw = 1, delay = '1ms' )
self.addLink( h24, s4, bw = 1, delay = '1ms' )
```

Pada simulasi terakhir adalah mengkondisikan semua traffic dari S3 ke ISP maupun ke Akamai Google di DROP maka hasilnya selalu time out seperti pada gambar 4.15. Untuk menjalankan perintah ini adalah dengan menghapus aturan add-flow atau merubah actionnya menjadi DROP.

```
mininet> sh ovs-ofctl del-flow s2 action=normal

atau

mininet> sh ovs-ofctl add-flow s2 action=drop
```

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari uraian yang telah dipaparkan pada bab-bab sebelumnya, maka dapat diambil kesimpulan:

1. Simulator mininet adalah tool yang tepat untuk beradaptasi dengan teknologi jaringan komputer berbasis SDN. Lingkungan testbed yang dibangun ini mampu memberikan pandangan bahwa teknologi jaringan komputer telah bergeser dari yang bersifat statis menjadi dinamis.
2. Pada lingkungan jaringan berbasis SDN, dimungkinkan admin network melakukan setup perangkat tidak perlu mensetup per perangkat, tetapi bisa dengan sebuah controller untuk beberapa perangkat.
3. Dengan bergesernya jaringan komputer ke basis SDN maka diperlukan tidak hanya network admin tetapi juga network programmer.
4. Pada teknologi jaringan berbasis SDN, memungkinkan pengguna untuk membuat/modifikasi isi dari perangkat tersebut.
5. Teknologi jaringan berbasis SDN saat ini banyak menawarkan aplikasi switch yang bersifat opensource seperti Floodlight, POX, Open Vswitch.

5.2 Saran

Beberapa saran yang dapat disampaikan oleh penulis terutama bagi pihak yang hendak mengembangkan sistem serupa adalah sebagai berikut:

1. Pengumpulan data dari obyek yang diteliti sebaiknya sedetail mungkin dengan ditambah dokumentasi permasalahan yang sering terjadi.
2. Perlunya sistem pendukung untuk menjalankan simulator Mininet, sehingga proses generate dan analisis menjadi cepat.

DAFTAR PUSTAKA

- [1]. <https://github.com/Mininet/Mininet/wiki/Documentation> diakses pada 6 April 2014.
- [2]. <http://www.nsnam.org/overview/what-is-ns-3/> diakses pada 7 April 2014.
- [3]. <https://www.netacad.com/web/about-us/cisco-packet-tracer> diakses pada 20 Mei 2014.
- [4]. Bob Lantz dkk, *A Network in a Laptop: Rapid Prototyping for Software-Defined Networks*, 2010.
- [5]. Nick McKeown dkk, *OpenFlow: Enabling Innovation in Campus Networks*, 2008.
- [6]. Nikhil Handigol dkk, *Reproducible Network Experiments Using Container-Based Emulation*, 2012.
- [7]. ONF White Paper, *Software-Defined Networking: The New Norm for Networks*, 2012.
- [8]. Rob Sherwood, *Can the Production Network Be the Testbed?*, 2007.

LAMPIRAN-LAMPIRAN

akakomOK-real-topo.py

```
"""Custom topology example
```

Two directly connected switches plus a host for each switch:

```
host --- switch --- switch --- host
```

Adding the 'topos' dict with a key/value pair to generate our newly defined topology enables one to pass in '--topo=mytopo' from the command line.

```
from mininet.topo import Topo
from mininet.link import TCLink
```

```
class MyTopo( Topo ):
    "Simple topology example."
```

```
    def __init__( self ):
        "Create custom topo."
```

```
        # Initialize topology
        Topo.__init__( self )
```

```
        # Add hosts and switches
        h1 = self.addHost( 'h1' )
        h2 = self.addHost( 'h2' )
        h3 = self.addHost( 'h3' )
        h4 = self.addHost( 'h4' )
        h11 = self.addHost( 'h11' )
        h12 = self.addHost( 'h12' )
        h13 = self.addHost( 'h13' )
        h14 = self.addHost( 'h14' )
        h21 = self.addHost( 'h21' )
        h22 = self.addHost( 'h22' )
        h23 = self.addHost( 'h23' )
        h24 = self.addHost( 'h24' )
        h100 = self.addHost( 'h100' ) #Internet Service Provider
        h101 = self.addHost( 'h101' ) #Akamai Google
        s2 = self.addSwitch( 's2' )
        s3 = self.addSwitch( 's3' )
        s4 = self.addSwitch( 's4' )
```

```
    # Add links
    self.addLink( h100, s2, bw = 13, delay = '5ms' )
    self.addLink( h101, s2, bw = 15, delay = '10ms' )
    self.addLink( h1, s2, bw = 1000, delay = '1ms' )
    self.addLink( h2, s2, bw = 1000, delay = '1ms' )
    self.addLink( h3, s2, bw = 100, delay = '1ms' )
    self.addLink( h4, s2, bw = 100, delay = '1ms' )
    self.addLink( s3, s2, bw = 100, delay = '3ms' )
    self.addLink( h11, s3, bw = 100, delay = '1ms' )
    self.addLink( h12, s3, bw = 100, delay = '1ms' )
    self.addLink( h13, s3, bw = 100, delay = '1ms' )
    self.addLink( h14, s3, bw = 100, delay = '1ms' )
    self.addLink( s4, s2, bw = 100, delay = '3ms' )
    self.addLink( h21, s4, bw = 100, delay = '1ms' )
    self.addLink( h22, s4, bw = 100, delay = '1ms' )
    self.addLink( h23, s4, bw = 100, delay = '1ms' )
    self.addLink( h24, s4, bw = 100, delay = '1ms' )
```

```
topos = { 'mytopo': ( lambda: MyTopo() ) }
```

akakomOK.py

```
# Add hosts and switches
```



```

h1 = self.addHost( 'h1' )
h2 = self.addHost( 'h2' )
h3 = self.addHost( 'h3' )
h4 = self.addHost( 'h4' )
h11 = self.addHost( 'h11' )
h12 = self.addHost( 'h12' )
h13 = self.addHost( 'h13' )
h14 = self.addHost( 'h14' )
h21 = self.addHost( 'h21' )
h22 = self.addHost( 'h22' )
h23 = self.addHost( 'h23' )
h24 = self.addHost( 'h24' )
h100 = self.addHost( 'h100' ) #Internet Service Provider
h101 = self.addHost( 'h101' ) #Akamai Google
s2 = self.addSwitch( 's2' )
s3 = self.addSwitch( 's3' )
s4 = self.addSwitch( 's4' )

```

```

# Add links
self.addLink( h100, s2, bw = 13, delay = '5ms' )
self.addLink( h101, s2, bw = 15, delay = '10ms' )
self.addLink( h1, s2, bw = 2, delay = '1ms' )
self.addLink( h2, s2, bw = 2, delay = '1ms' )
self.addLink( h3, s2, bw = 2, delay = '1ms' )
self.addLink( h4, s2, bw = 1, delay = '1ms' )
self.addLink( s3, s2, bw = 100, delay = '3ms' )
self.addLink( h11, s3, bw = 1, delay = '1ms' )
self.addLink( h12, s3, bw = 1, delay = '1ms' )
self.addLink( h13, s3, bw = 1, delay = '1ms' )
self.addLink( h14, s3, bw = 1, delay = '1ms' )
self.addLink( s4, s2, bw = 100, delay = '3ms' )
self.addLink( h21, s4, bw = 1, delay = '1ms' )
self.addLink( h22, s4, bw = 1, delay = '1ms' )
self.addLink( h23, s4, bw = 1, delay = '1ms' )
self.addLink( h24, s4, bw = 1, delay = '1ms' )

```

```

topos = { 'mytopo': ( lambda: MyTopo() ) }

```

LAMPIRAN 1

SUSUNAN ORGANISASI TIM PENELITIAN DAN PEMBAGIAN TUGAS

No	Nama/NIDN	Instansi Asal	Bidang Ilmu	Alokasi waktu (Jam/Minggu)	Uraian Tugas
1	Yagus Cahyadi S.T., M.Eng 0518128101	STMIK AKAKOM	<i>Cloud Computing</i> dan <i>Web</i>	2	Identifikasi masalah, analisa dan design, implementasi dan simulasi
2	Agung Budi Prasetyo S.Kom., M.Kom 0003087106	STMIK AKAKOM	Rekayasa perangkat lunak	2	Identifikasi data pendukung, pengujian dan evaluasi

LAMPIRAN 2
BIODATA KETUA DAN ANGGOTA

Ketua Peneliti

A. Identitas Diri

1	Nama Lengkap (dengan gelar)	Yagus Cahyadi S.T., M.Eng
2	Jenis Kelamin	Laki – Laki
3	Jabatan Fungsional	Tenaga Pengajar
4	NIP/NIK/Identitas lainnya	121175
5	NIDN	0518128101
6	Tempat dan Tanggal Lahir	Ngawi, 18 Desember 1981
7	Email	yagus.cahyadi@akakom.ac.id
8	Nomor Telepon/HP	085725720897
9	Alamat Kantor	Jl. Raya Janti Karangjambe 143 Yogyakarta
10	Nomor Telepon/Faks	(0274) 486664/486438
11	Lulusan Yang Telah Dihasilkan	S-1 = 0 Orang, S-2 = 0 Orang, S-3 = 0 Orang
12	Mata Kuliah Yang Diampu	1. Pengenalan Web (HMTL) 2. Pemrograman 1 (PHP) 3. Cloud Computing (OpenShift dan PHP) 1. Enterprise Information Sistem

B. Riwayat Pendidikan

	S1	S2
Nama Perguruan Tinggi	UGM	Asia University Taiwan
Bidang Ilmu	Teknik Elektro	Computer Science
Tahun Masuk-Lulus	2001 - 2007	2010 - 2012
Judul Skripsi/Tesis/Disertasi	Image Index dan Query pada Aplikasi Internet TV	An LTE Handover Algorithm Based GM (1, N) in High Speed Railway
Nama Pembimbing/Promotor	Dr. Selo Sulistyio	Dr. Hsing-Chung Chen

C. Pengalaman Penelitian Dalam 5 Tahun Terakhir (Bukan Skripsi, Tesis maupun Disertasi)

No	Tahun	Judul Penelitian	Pendanaan	
			Sumber*	Jml (Juta Rp)
1	2012	A Grey Prediction Based Hard Handover Hysteresis Algorithm for 3GPP LTE System	Asia University Grant	-
2	2012	Application of GM (1,1) Model for Forecasting Research Trends of Security in Internet of Things	Asia University Grant	-
3	2012	Using GM (1, 1) Model to Forecast the Trend of Research in Internet of Things	Asia University Grant	-

D. Pengalaman Pengabdian Kepada Masyarakat dalam 5 Tahun Terakhir

No	Tahun	Judul Pengabdian Kepada Masyarakat	Pendanaan	
			Sumber*	Jml (Juta Rp)
1	2013	Sosialisasi LJK SMA Pancasila Purworejo	AKAKOM	3.000.000
2	2013	Diklat Operator Komputer Pemda Bantul Tahun 2013	PPM STMIK AKAKOM	500.000

E. Publikasi Artikel Ilmiah Dalam Jurnal Dalam 5 Tahun Terakhir

No	Judul karya Ilmiah	Nama Jurnal	Tahun
1	A Grey Prediction Based Hard Handover Hysteresis Algorithm for 3GPP LTE System	Seventh International Conference on Broadband and Wireless Computing, Communication and Applications, Canada	2012
2	Application of GM (1,1) Model for Forecasting Research Trends of Security in Internet of Things	The 5th IET International Conference on Ubi-media Computing, Xining, China	2012
3	Using GM (1, 1) Model to Forecast the Trend of Research in Internet of Things	International Conference on Automatic Control and Artificial Intelligence, The Institution of Engineering and Technology (IET), Xiamen, China	2012

F. Pemakalah Seminar Ilmiah (Oral Presentation) Dalam 5 Tahun Terakhir

No	Nama Pertemuan Ilmiah/ Seminar	Judul Artikel Ilmiah	Waktu Dan Tempat
1	-	-	-

G. Karya Buku Dalam 5 Tahun Terakhir

No	Judul Buku	Tahun	Jumlah Halaman	Penerbit
1	-	-	-	-

H. Perolehan HKI Dalam 5-10 Tahun Terakhir

No	Judul/Tema HKI	Tahun	Jenis	Nomor P/ID
1	-	-	-	-

I. Pengalaman Merumuskan Kebijakan Publik/Rekayasa Sosial Lainnya dalam 5 Tahun Terakhir

No	Judul/Tema/Jenis Rekayasa Sosial Lainnya yang telah diterapkan	Tahun	Tempat penerapan	Respon Masyarakat
1	-	-	-	-

J. Penghargaan Dalam 10 Tahun Terakhir (dari Pemerintah, Asosiasi atau institusi lainnya)

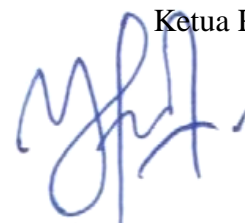
No	Jenis Penghargaan	Institusi Pemberi Penghargaan	Tahun
1	-	-	-

Semua data yang saya isikan dan tercantum dalam biodata ini adalah benar dan dapat dipertanggung jawabkan secara hukum. Apabila dikemudian hari ternyata dijumpai ketidaksesuaian dengan kenyataan, saya sanggup menerima sanksi.

Demikian biodata ini saya buat dengan sebenarnya untuk memenuhi salah satu persyaratan dalam pengajuan Penelitian DIPA Kopertis V.

Yogyakarta, 19 Mei 2014

Ketua Peneliti



(Yagus Cahyadi S.T., M.Eng)

Anggota Peneliti

A. Identitas Diri

1.	Nama Lengkap (dengan gelar)	Agung Budi Prasetyo, S.Kom., M.Kom.
2.	Jenis Kelamin	Laki
3.	Jabatan Fungsional	Asisten Ahli
4.	NIP/NIK/Identitas lainnya	197108032005011001
5.	NIDN	0003087106
6.	Tempat dan Tanggal Lahir	Jakarta, 3 Agustus 1971
7.	E-mail	agung_bp@akakom.ac.id
8.	Nomor Telepon/HP	08156880231
9.	Alamat Kantor	Jln. Raya Janti No 143 Yogyakarta
10.	Nomor Telepon/Faks	(0274) 486664 / 486438
11.	Lulusan yang Telah Dihasilkan	S-1 = 98 orang; S-2 = - orang; S-3 = -orang
12.	Mata Kuliah yang Diampu	1. Algoritma dan Pemrograman
		2. Pemrograman Web
		3. Struktur Data
		4. Pemrograman Berorientasi Obyek
		5. Pengenalan Teknologi Informasi

B. Riwayat Pendidikan

	S-1	S-2	S-3
Nama PT	Universitas Kristen Duta Wacana	Universitas Gadjah Mada	-
Bidang Ilmu	Teknik Informatika	Ilmu Komputer	-
Tahun Masuk-Lulus	1990-1996	2001-2003	-
Judul Skripsi/Tesis/Disertasi	Program Visualisasi Eliminasi Gaus pada Sparse Matrix	Implementasi Sistem Kendali Berbasis Komputer Untuk Motor Penggerak Alat Pengunci pada Pintu Rumah	-
Nama Pembimbing /Promotor	Dr./Ir.F.Soesianto,B.Sc	Drs.Jazi Eko,Ph.D	-

C. Pengalaman Penelitian dalam 5 Tahun Terakhir

No	Tahun	Judul Penelitian	Pendanaan	
			Sumber	Jml (Juta Rp)
1	2009	Search Engine untuk Mencari File dengan Deteksi Header	Puslitbang & PPM STMIK AKAKOM	2.5
2	2011	Implementasi Circular Linked List pada Pembuatan Game Permainan Tradisional Congklak	Puslitbang & PPM STMIK AKAKOM	2.5
3	2013	Rekayasa Permainan Congklak Digital, Sebuah Upaya Alternatif Untuk Menjaga Kelestarian Seni Permainan Rakyat Tradisional di Indonesia	Dikti	13.5

D. Pengalaman Pengabdian Kepada Masyarakat dalam 5 Tahun Terakhir

No.	Tahun	Judul Pengabdian Kepada Masyarakat	Pendanaan	
			Sumber	Jml (Juta Rp)
1	2008	Pengenalan Internet Pada Anak Di Domby Kid's Hope (DKH) Jogjakarta	Puslitbang & PPM STMIK AKAKOM	0.2
2	2010	Kegiatan Tanggap Bencana Merapi Dalam Bentuk Penyediaan Informasi Online Dan Pendistribusian Bantuan Material	Swadaya	20
3	2010	P emuktahiran data untuk proses e-KTP Kabupaten Bantul 2009	PPM STMIK AKAKOM	0.2
4	2011	Pelatihan Internet Bagi Anggota Umkm Se Propinsi Diy	PPM STMIK AKAKOM	0.5
5	2012	Pelatihan MS OFFICE 2007 bagi para Bidan Dinas Kesehatan Kabupaten Bantul, DIY	PPM STMIK AKAKOM	0.5
6	2013	Diklat Operator Komputer Pemda Bantul Tahun 2013	PPM STMIK AKAKOM	0.5

E. Publikasi Artikel Ilmiah dalam Jurnal dalam 5 Tahun Terakhir

No.	Judul Artikel Ilmiah	Nama Jurnal	Volume/ Nomor/Tahun
1	Search Engine untuk Mencari File dengan Deteksi Header	FORMAT	Volume 10 Nomor 3, Edisi September 2009, ISSN : 1410-9158
2	File Extension Auto Repair Dengan Teknik Deteksi Header	Proceeding SRITI	Volume V 2010, ISSN : 1907-3526
3	Game Congklak Digital Sebagai Upaya Pelestarian Permainan Rakyat Tradisional	Proceeding STE	ISSN : 2354-6344

F. Pemakalah Seminar Ilmiah (*Oral Presentation*) dalam 5 Tahun Terakhir

No.	Nama Pertemuan Ilmiah / Seminar	Judul Artikel Ilmiah	Waktu dan Tempat
1	Seminar Nasional Riset Teknologi Informasi SRITI	File Extension Auto Repair Dengan Teknik Deteksi Header	2010, STMIK AKAKOM Yogyakarta
2	Seminar Nasional Teknik Elektro	Game Congklak Digital Sebagai Upaya Pelestarian Permainan Rakyat Tradisional	4 Desember 2013, Universitas Negeri Surabaya

Semua data yang saya isikan dan tercantum dalam biodata ini adalah benar dan dapat dipertanggung jawabkan secara hukum. Apabila dikemudian hari ternyata dijumpai ketidaksesuaian dengan kenyataan, saya sanggup menerima sanksi.

Demikian biodata ini saya buat dengan sebenarnya untuk memenuhi salah satu persyaratan dalam pengajuan Penelitian DIPA Kopertis V.

Yogyakarta, 19 Mei 2014

Yang menyatakan

(Agung Budi Prasetyo, S.Kom., M.Kom)

LAMPIRAN 3
SURAT PERNYATAAN KETUA PENELITI

YAYASAN PENDIDIKAN WIDYA BAKTI
STMIK AKAKOM
Y O G Y A K A R T A

mencetak programmer bersertifikasi

Yang Pertama dan Utama
www.akakom.ac.id

143, Karang Jambe, Yogyakarta 55198 Telp. (0274) 486664, Fax (0274) 486438 e-mail: info@akakom.ac.id

SURAT PERNYATAAN KETUA PENELITI/PELAKSANA

Yang bertanda tangan di bawah ini:

Nama : Yagus Cahyadi S.T., M.Eng
NIDN : 0518128101
Pangkat/Gol : -
Jabatan Fungsional : Tenaga Pengajar

Dengan ini menyatakan bahwa proposal penelitian saya dengan judul:

“Implementasi Simulator *Mininet* Sebagai Pendukung Jaringan Komputer Masa Depan Berbasis *Software Defined Network* di STMIK AKAKOM”

Yang diusulkan dalam skema Penelitian DIPA Kopertis V untuk tahun anggaran 2014 bersifat original dan belum pernah dibiayai oleh lembaga/sumber dana lain.

Bila mana dikemudian hari ditemukan ketidaksesuaian dengan pernyataan ini, maka saya bersedia dituntut dan diproses sesuai dengan ketentuan yang berlaku dan mengembalikan seluruh biaya penelitian yang sudah diterima ke kas Negara. Demikian pernyataan ini dibuat dengan sesungguhnya dan dengan sebenar-benarnya.

Yogyakarta, 19 Mei 2014

Mengetahui,

Yang menyatakan,

Ketua Lembaga Penelitian



Dra. Svamsu Windarti, M.T., Apt.

NIP/NIK 1966071019932001



Yagus Cahyadi, S.T., M.Eng

NIP 121175

**LAPORAN KEUANGAN
PENELITIAN DIPA KOPERTIS WILAYAH V**



**JUDUL PENELITIAN :
IMPLEMENTASI SIMULATOR *MININET* SEBAGAI PENDUKUNG
JARINGAN KOMPUTER MASA DEPAN BERBASIS *SOFTWARE*
DEFINED NETWORK DI STMIK AKAKOM**

Dibiayai oleh :
Dana Penelitian DIPA Kopertis V Tahun anggaran 2014
Kopertis Wilayah V
Daerah Istimewa Yogyakarta
Sesuai dengan Surat Perjanjian Pelaksanaan Bantuan Penelitian Tahun Anggaran 2014
Nomor : 2174/K6/KM/2014

Tahun ke 1 dari rencana 1 tahun
Ketua/ Anggota Tim
Yagus Cahyadi S.T., M.Eng
NIDN - 0518128101
Agung Budi Prasetyo S.Kom., M.Kom
NIDN - 0003087106

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
AKAKOM
YOGYAKARTA
Nopember 2014**

Rekapitulasi Penggunaan Dana Penelitian

Judul : Implementasi Simulator *Mininet* Sebagai Pendukung Jaringan Komputer Masa Depan berbasis *Software Defined Network* di STMIK AKAKOM

Skema Hibah : Penelitian dengan Dana DIPA Kopertis V Tahun anggaran 2014

Peneliti / Pelaksana

Nama Ketua : Yagus Cahyadi S.T., M.Eng.

Perguruan Tinggi : STMIK AKAKOM

NIDN : 0518128101

Nama Anggota (1) : Agung Budi Prasetyo, S.Kom, M.Kom.

Tahun Pelaksanaan : Tahun ke 1 dari rencana 1 tahun

Dana Tahun Berjalan : Rp.4.000.0000,-

Dana Mulai Diterima Tanggal : 2014-07-18

Rincian Penggunaan

1. HONOR OUTPUT KEGIATAN

Item Honor	Volume	Satuan	Honor/Jam (Rp)	Total (Rp)
Ketua Peneliti	64	jam	Rp 11,250.00	Rp 720,000.00
Anggota	64	jam	Rp 7,500.00	Rp 480,000.00
Sub Total				Rp 1,200,000.00

2. BELANJA BAHAN

Item	Volume	Satuan	@	Total (Rp)
Paket Raspberrypi	1	paket	Rp 814,000.00	Rp 814,000.00
Kabel HDMI 1,5	1	unit	Rp 40,000.00	Rp 40,000.00
Monitor Samsung	1	unit	Rp 1,075,000.00	Rp 1,075,000.00
IM3 25	1	paket	Rp 25,500.00	Rp 25,500.00
Hardisk Toshiba	1	unit	Rp 421,000.00	Rp 421,000.00
Mouse Logic M100R	1	unit	Rp 59,000.00	Rp 59,000.00
Keyboard Logitech	1	unit	Rp 94,000.00	Rp 94,000.00
Sub Total				Rp 2,528,500.00

3. BELANJA BARANG NON OPERASIONAL LAINNYA

Item	Volume	Satuan	@	Total (Rp)
ATK & Meterai	1	paket	Rp 117,500.00	Rp 117,500.00

Fotocopy & Jilid Proposal	4	paket	Rp 36,275.00	Rp 36,275.00
Fotocopy & Jilid Laporan	4	paket	Rp 49,400.00	Rp 49,400.00
Buku Referensi	1	eksp	Rp 76,000.00	Rp 76,000.00
Sub Total				Rp 279,175.00
Total Pengeluaran Dalam Satu Tahun				Rp 4,007,675.00

Mengetahui
Kepala Puslitbang dan PPM



(Dra. Syamsu Windarti, M.T)
NIP/NIK 1966071019932001


Yogyakarta, 10 Nopember 2014,
Ketua ,

(Yagus Cahyadi S.T., M.Eng.)
NIP/NIK 121175

Untuk Bulan Juli S/D November 2014
 Penelitian DIPA Kopertis
 Implementasi Simulator Mininet Sebagai Pendukung Jaringan Komputer Masa Depan berbasis Software Defined Network di STMIK
 AKAKOM
 Sesuai dengan Surat Perjanjian Pelaksanaan Bantuan Penelitian Tahun Anggaran 2014 Nomor : 2174/K6/KM/2014

No	Nama	Gol	Tugas dalam Kegiatan	Jumlah Jam	HR (Rp/Jam)	Jumlah Bruto	PPh Ps.21	Penerimaan Bersih	NPWP	Tanda-tangan
1	Yagus Cahyadi	IIIa	Ketua Peneliti	64	Rp 11,250	Rp 720,000	Rp 36,000.00	Rp 684,000	597509876646000	
2	Agung Budi Prasetyo	IIIa	Anggota Peneliti	64	Rp 7,500	Rp 480,000	Rp 24,000.00	Rp 456,000	476812300542000	
Jumlah						Rp 1,200,000	Rp 60,000.00	Rp 1,140,000		

Yogyakarta, 10 Nopember 2014,
Ketua Peneliti,


 (Yagus Cahyadi S.T., M.Eng.)
 NIP/NIK 121175



KEMENTERIAN KEUANGAN R.I.
DIREKTORAT JENDERAL PAJAK

**SURAT SETORAN PAJAK
(SSP)**

LEMBAR

1

Untuk Arsip Wajib Pajak

NPWP : 59 750 987 6 646 000

Diisi sesuai dengan Nomor Pokok Wajib Pajak yang dimiliki

NAMA WP : YAGUS CAHYADI, S.T., M.Eng

ALAMAT WP : DS. SUKOWIJONO 1 RT 05 RW 04 SUKOWIJONO PADAS
NGAWI

NOP :

Diisi sesuai dengan Nomor Objek Pajak

ALAMAT OP :

Kode Akun Pajak

4 1 1 1 2 1

Kode Jenis Setoran

1 0 0

Uraian Pembayaran : Pph Pasal 21 sebesar 5%
dari honor penelitian RIPA seperti 5
sebesar Rp 720.000,-

Masa Pajak

Jan	Feb	Mar	Apr	Mei	Jun	Jul	Ags	Sep	Okt	Nov	Des
								X			

Beri tanda silang (X) pada kolom bulan, sesuai dengan pembayaran untuk masa yang berkenaan

Tahun Pajak

2014

Diisi Tahun terutangnya Pajak

Nomor Ketetapan :

Diisi sesuai Nomor Ketetapan : STP, SKPKB atau SKPKBT

Jumlah Pembayaran : Rp 36.000,- Diisi dengan rupiah penuh

Terbilang : TIGA RULUH ENAM RIBU RUPIAH

Diterima oleh Kantor Penerima Pembayaran

Tanggal

Cap dan tanda tangan

Nama Jelas :

Wajib Pajak/Penyetor

YOGYAKARTA, Tanggal 5 SEPTEMBER 2014

Cap dan tanda tangan

Nama Jelas : YAGUS CAHYADI

"Terima Kasih Telah Membayar Pajak - Pajak Untuk Pembangunan Bangsa"
Ruang Validasi Kantor Penerima Pembayaran



KEMENTERIAN KEUANGAN R.I.
DIREKTORAT JENDERAL PAJAK

SURAT SETORAN PAJAK (SSP)

LEMBAR

1

Untuk Arsip Wajib Pajak

NPWP : 47 681 230 0 542 000

Diisi sesuai dengan Nomor Pokok Wajib Pajak yang dimiliki

NAMA WP : AGUNG BUDI PRASETYO, S.KOM, M.KOM

ALAMAT WP : KOMPLEKS MUTIARA SAMBISARI A5 SAMBISARI PURWOMARTANI
KALASAN CEMAN YOGYAKARTA

NOP :

Diisi sesuai dengan Nomor Objek Pajak

ALAMAT OP :

Kode Akun Pajak

411121

Kode Jenis Setoran

100

Uraian Pembayaran : PPh Pasal 21 sebesar 5%
Honor Penelitian DIPA KOPERTIS 5
Sebesar Rp 480.000,-

Masa Pajak

Jan	Feb	Mar	Apr	Mei	Jun	Jul	Ags	Sep	Okt	Nov	Des
								X			

Beri tanda silang (X) pada kolom bulan, sesuai dengan pembayaran untuk masa yang berkenaan

Tahun Pajak

2014

Diisi Tahun terutangnya Pajak

Nomor Ketetapan :

Diisi sesuai Nomor Ketetapan : STP, SKPKB atau SKPKBT

Jumlah Pembayaran : Rp 24.000,-

Diisi dengan rupiah penuh

Terbilang : DUA PULUH EMPAT RIBU RUPIAH

Diterima oleh Kantor Penerima Pembayaran

Tanggal

Cap dan tanda tangan

Nama Jelas :

Wajib Pajak/Penyetor

YOGYAKARTA, Tanggal 5 SEPTEMBER 2014

Cap dan tanda tangan

Nama Jelas : YAGUS CANYADI

"Terima Kasih Telah Membayar Pajak - Pajak Untuk Pembangunan Bangsa"
Ruang Validasi Kantor Penerima Pembayaran

DEPARTEMEN KEUANGAN RI
DIREKTORAT JENDERAL PAJAK
KANTOR PELAYANAN PAJAK

118154081

BUKTI PENERIMAAN NEGARA

NTPN : 0005150904100105
NTP : 55165A-02/14/011678
NPWP : 59.750.987.6 646000
Nama WP : YAGUS CAHYADI
Alamat WP : DN SUKOWIYONO I
Kota : NGAWI
Jml Pembayaran: Rp. 36,000
(TIGA PULUH ENAM RIBU RUPIAH)

Kd Akun Pajak: 411121
KdJns Setoran: 100
Masa Pajak : 09 - 09 - 2014

No.Ketetapan : 0000000000000000

Diterima oleh Kantor Penerima pembayaran
Tanggal : 06-09-2014
Petugas Loker

Wajib Pajak / Penyetor
Tanggal: 06-09-2014*

FEBRIASTUTI W
Nippos : 313002991

Syarat dan ketentuan berlaku

KANTOR POS



Lacak status : <http://www.posindonesia.co.id>

PT.PBT 06/2014

DEPARTEMEN KEUANGAN RI
DIREKTORAT JENDERAL PAJAK
KANTOR PELAYANAN PAJAK

118154082

BUKTI PENERIMAAN NEGARA

NTPN : 0301070412000707
NTP : 55165A-02/14/011679
NPWP : 47.681.230.0 542000
Nama WP : AGUNG BUDI PRASETYO
Alamat WP : PERUM APH A-5 SETURAN NO., CATURTUNGAL
Kota : SLEMAN
Jml Pembayaran: Rp. 24,000
(DUA PULUH EMPAT RIBU RUPIAH)

Kd Akun Pajak: 411121
KdJns Setoran: 100
Masa Pajak : 09 - 09 - 2014

No.Ketetapan : 0000000000000000

Diterima oleh Kantor Penerima pembayaran
Tanggal : 06-09-2014
Petugas Loker

Wajib Pajak / Penyetor
Tanggal: 06-09-2014*

FEBRIASTUTI W
Nippos : 313002991

Syarat dan ketentuan berlaku

KANTOR POS



Lacak status : <http://www.posindonesia.co.id>

PT.PBT 06/2014

[illegible]

Jl. Kusuma Negara No. 97 Telp. (0274) 8283029
Jl. Colombo 2A Yogyakarta , (0274)555-451
Jl. Laksda Adisutjipto Km 6 No. 66 Yogyakarta, (0274)7158953
Jl. Ambarukmo No. 63 Yogyakarta, (0274)3287483
Jl. Hos. Cokroaminoto No. 64 Yogyakarta, (0274) 619 818

Banyaknya	Keterangan	Harga	Jumlah
93	FC 70 / 80 / 100	175	16.275
	FC A3 / B4 / WR		
4	2ind	5000	20.000
TOTAL			36.275



Printed : 0 x

Printed by : RETNA

Nota Penjualan

Nomor : JL-YK-0231033
 Ref :
 Gudang : G-YK
 Sales : SCJ-YOGA
 Tanggal : 12/08/2014 17:01:46
 Term (hari) : 0
 Mata Uang : IDR
 Kurs : 1.00
 Customer : YK-CASH-PC
 Alamat :
 CASH PC

Barang	Deskripsi	Qty	Satuan	Harga	Pajak	Diskon	Sub Total
PC-HD-IN-TOS-2.5"-0250	Hardisk Toshiba Int 2.5" 250 GB (1 th)	1	PCS	421,000.00	0	0	421,000.
PC-ME-LOG-M100R	Mouse Logitech M100R (1 th)	1	PCS	59,000.00	0	0	59,000.
PC-KB-LOG-K120	Keyboard Logitech K120 (1 th)	1	UNIT	94,000.00	0	0	94,000.

Ketersangan :
 Total : 574,000.
 Diskon : 0.
 Total Keseluruhan : 574,000.

UNAS
 Penjualan 12 AUG 2014 Gudang



Hormat kami
 (SCJ-YOGA)

Printed : 0 x

Printed by : RETNA

12/08/2014 17:01:46

TOKO ALAT TULIS & ASSESORIES
PRIMA JAYA
PURWOMARTANI - YOGYAKARTA

NO Nota :

KODE	NAMA BARANG	HARGA	JUMLAH
5	Meterai 6000	6500	32.500
1	A4 SK	34.000	34.000
1	F4 SK	36.000	36.000
1	Buku SSP	15.000	15.000
TGL. 5/8/14		TOTAL HARGA	117.500

Barang yang sudah dibeli tidak dapat ditukar/dikemb.

TOKO ALAT TULIS & ASSESORIES
PRIMA JAYA
PURWOMARTANI - YOGYAKARTA

Barokah Photo Copy
Print - Scan - Photo Cpy
Telp. 081 319315427

Yogyakarta,/...../20
Sopalan, Maguwoharjo, Slem
Depan Kantor SA

NOTA No. **Cano**

BANYAKNYA	NAMA BARANG	HARGA	JUMLAH
168	Photo Copy A4	175	29.400
	Photo Copy F4		
	Photo Copy A3		
	Laminating A4		
	Laminating F4		
	Laminating A3		
4	Jilid A4	5000	20.000
	Jilid F4		
	Print A4		
	Print F4		
	Print A3		
Jumlah Rp.			49.400

Tanda Terima

SOPALAN, MAGUWOHARJO, SLEMAN
Kantor SA

Tanggal : 10 - 8 - 2014
Kd. Yn :

YAMA PUTRA
Komplek Taman Pintar,
Lantai 2, No 77 Yogyakarta
Tlp : 0274-6573688

No.	Nama Buku	Harga	Jumlah
	Peringatan Kompuh Wazqib		70.000
Jumlah Rp.			70.000

Barang yang sudah dibeli tidak dapat ditukar/dikemb.

